

(19) 日本国特許庁 (JP)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-170634

(43) 公開日 平成11年(1999)6月29日

(51) IntCl.⁶
B 41 J 5/30
G 06 F 3/12
7/00
G 06 T 1/20
H 04 N 1/40

識別記号

F I
B 41 J 5/30 Z
G 06 F 3/12 L
H 04 N 1/411
G 06 F 7/00 A
15/66 L

審査請求 未請求 請求項の数1 OL (全30頁) 最終頁に続く

(21) 出願番号 特願平10-168202

(22) 出願日 平成10年(1998)6月16日

(31) 優先権主張番号 877, 345

(32) 優先日 1997年6月17日

(33) 優先権主張国 米国(US)

(71) 出願人 398038580

ヒューレット・パッカード・カンパニー
HEWLETT-PACKARD COMPANY

アメリカ合衆国カリフォルニア州パロアルト
ハノーバー・ストリート 3000

(72) 発明者 ゲーリー・エル・ヴァンドラン・ジュニア
アメリカ合衆国01890マサチューセッツ州
ウインチエスター、フェアモント・ストリート 25

(74) 代理人 弁理士 岡田 次生

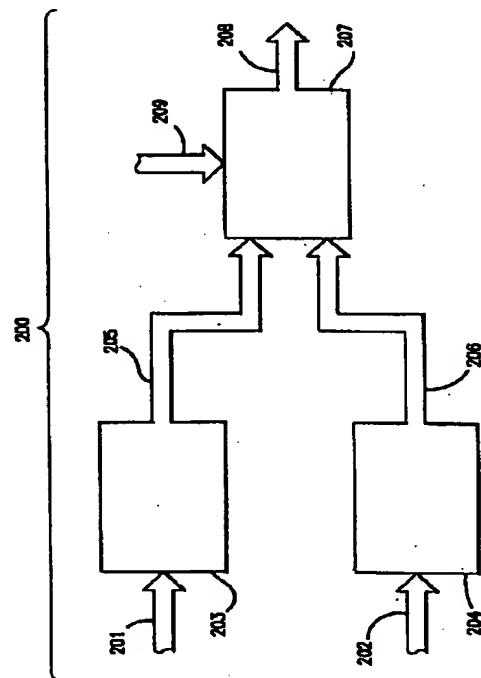
最終頁に続く

(54) 【発明の名称】 データ処理パイプライン

(57) 【要約】

【課題】 印刷データのデータ要素の圧縮／伸張、カラー空間変換、中間調処理などの処理を最適化する。

【解決手段】 ソースストリームをテキスト、線画、グラフィックス、イメージ等のデータ要素の特性に基づいて複数のデータストリームに分割する。分割は、無損失性(可逆)と損失性(非可逆)のピクセルデータに分割することができる。各データストリームは、データ処理パイプラインにおいて、それぞれ別個に圧縮、カラー空間変換、伸張等の処理を受ける。処理を受けたデータストリームは併合ユニットで併合され、オリジナルの画像を再構成し、必要ならば中間調処理を受け、プリントエンジンに渡される。



【特許請求の範囲】

【請求項1】複数のデータ要素からそれぞれ形成される複数のデータストリームを別個に処理するためのデータ処理パイプラインであって、前記複数のデータストリームにより定められ、前記データ要素に対応する併合データ要素で形成される併合データストリームを使用し、前記複数のデータストリームの第1のデータストリームを受け取る第1の入力を有し、該第1のデータストリームの前記データ要素について第1の変換を実行し、第1の変換されたデータストリームを生成する第1のパイプライン処理ユニットと、前記複数のデータストリームの第2のデータストリームを受け取る第2の入力を有し、該第2のデータストリームの前記データ要素について第2の変換を実行し、第2の変換されたデータストリームを生成する第2のパイプライン処理ユニットと、前記第1の変換されたデータストリーム、前記第2の変換されたデータストリームおよび前記併合データストリームを受け取るよう構成され、前記第1の変換されたデータストリームおよび前記第2の変換されたデータストリームを、前記併合データストリームを使用して出力データストリームに併合する併合ユニットと、を備えるデータ処理パイプライン。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データの処理に関し、より具体的にはプリンタにおける印刷データ処理パイプラインの構成に関する。

【0002】

【従来の技術】プリンタの印刷データパイプラインは、印刷に備えてパイプラインに入力される印刷データについて多くの操作を実行する。これらの操作は、印刷データの圧縮、印刷データの伸張、カラー空間の変換、中間調処理(halftoning)を含む。実行される操作の種類およびその操作が実行される特定の順番は、パイプラインに入る印刷データの種類、プリントエンジンの機能、プリンタの使用できるメモリに依存して変化する。パイプラインに入ることができる印刷データの種類には、テキスト、線画(line art)、イメージ、グラフィックスを含むことができる。従来技術におけるパイプラインの実現は、多様な処理操作がファームウェアの制御下のプロセッサにより実行される。パイプラインに入る印刷データの種類に依存して、多くの可能なファームウェア・ルーチンが必要に応じて実行され、前述の操作を行う。

【0003】実行される印刷データ圧縮の操作の詳細は、パイプラインに入る印刷データの種類に依存する。たとえば、イメージ印刷データのような印刷データのある種類では、情報の何らかの損失(loss)をもたらす印刷データ圧縮ルーチンが許容されている。これらの印刷データの種類では、印刷された出力の品質低下は認められ

ない。印刷された出力で情報の損失が認められない圧縮ルーチンは、「視覚的に無損失性(visually lossless)」なシステムと呼ばれる。しかし、テキストや線画のような他の印刷データの種類についての印刷された出力の品質は、使用する印刷データ圧縮ルーチンが情報の損失をもたらさないことが重要である。

【0004】データ圧縮/伸張システムは、当該技術分野において既知であり、デジタルデータ信号のストリームを圧縮デジタルコード信号に符号化し、圧縮デジタルコード信号を最初のデータに復号化して戻す。データ圧縮は、所与の形式のデータを最初より少ないスペースで足りる代わりの形式に変換しようとする何らかの処理をいう。データ圧縮システムの目的は、デジタル情報の所与の内容を保持するのに必要な記憶容量の節約を行うことである。このデジタル情報がイメージやテキストのデジタル表示である場合には、データ圧縮システムが2つの一般的な型に分類される。すなわち、損失があるもの(lossy: 損失性)と無いもの(lossless: 無損失性)である。(なお、lossyおよびlosslessは、それぞれ非可逆および可逆ともいう。)

無損失性のシステムは、可逆(reciprocity)と呼ばれるもの有する。データ圧縮システムが可逆の特性(property)を持つためには、いかなる情報の変質または損失もなく、圧縮されたデータを再伸張または復号して最初の形に戻すことができなければならない。また復号化データと最初のデータは、互いに全く同一であり区別できないものでなければならない。よって可逆の特性は、情報論で使用される厳密にノイズ(雑音)が無いものと同義である。

【0005】いくつかのアプリケーションは、可逆の特性に厳密に準拠する必要がない。上に述べたように、このような特定のアプリケーションの1つが、イメージデータを扱う場合である。人間の眼はノイズに感知しやすいわけではないので、圧縮および伸張の間における情報の多少の変質または損失が許容される。情報のこの損失を与えるシステムを、「損失性データ圧縮システム」と呼ぶことにする。

【0006】データ圧縮システムの設計における重要な基準は、圧縮の効率性であり、これは圧縮比により特づけられる。圧縮比(compression ratio)は、圧縮形式のサイズで割った非圧縮形式のデータサイズの比である。データが圧縮可能であるためには、データは冗長度を含まなければならない。圧縮の効率性は、圧縮の手順が入力データの冗長度をいかに有効に使うかにより決定される。データが保管される通常のコンピュータでは、冗長度は個々の記号表示(たとえば、数字、バイト、文字)の均一でない使用および一連の記号のたびたび起こる発生(一般的な語、ブランクのレコードフィールドのような)の両方で起こる。

【0007】プリンタにより与えられ、受け入れられる

データレート(data rate)に関して、データ圧縮システムは十分なパフォーマンスを提供する。データが圧縮されるレートは、圧縮システムの入力データの処理レートにより決定される。達成されたデータレートを維持し、処理されたデータが使用可能でないことによる印刷の中止を防止するには、十分なパフォーマンスが必然的である。よって、データ圧縮伸張システムは、システム全体に悪影響を及ぼさないために十分なデータ帯域幅を有しなければならない。

【0008】通常、データ圧縮伸張システムのパフォーマンスは、圧縮および伸張に必要な計算、および統計データを保管するのに利用され圧縮処理を導くランダムアクセスメモリとこれに類するようなシステム要素のスピードにより限界づけられる。これは、圧縮伸張システムがファームウェアで実現される時に特に当てはまり、ファームウェアが一般用の中央処理ユニットを、データ圧縮伸張処理を実行するよう導く。このようなシステムでは、圧縮装置のパフォーマンスは、圧縮中の入力キャラクタあたり必要なプロセッサ・サイクル数により特性づけられる。サイクル数が小さければ小さいほど、パフォーマンスは良くなる。ファームウェアの解決方法は、ファームウェアの圧縮伸張スピードにより限界づけられる。これは、ファームウェアがいくつかの中央処理ユニット・サイクルを利用して各バイトを伸張するからである。よって、ファームウェア処理は通常、伸張スピードを上げるために圧縮比を落とすよう調整された。

【0009】一般用のデータ圧縮手順は、従来技術において既知である。3つの適切な手順、ハフマン法(Huffman)、タンストール法(Tunstall)、レンペルージブ法(Lempel-Ziv)がある。まず第一の一般用に開発されたデータ圧縮手順の1つが、ハフマン法である。簡単に述べると、ハフマン法は記号の全長のセグメントを可変長ワードにマップする。タンストール法は、記号の可変長セグメントを固定長バイナリワードにマップし、ハフマン法に準ずるものである。ハフマン手順と同様に、タンストール手順は、ソースデータの確率を前もって知っている必要がある。また、この前もって知っているという必要性は、データの統計的なストレンジス処理(statistic strength processing)を累積する適切なバージョンを利用することにより、ある程度までは満足することができる。

【0010】レンペルージブ手順は、記号の可変長セグメントを可変長バイナリワードにマップする。入力または出力セグメントに制約がない場合には漸近的に最適である。この手順では、入力データ列が、適切に成立したセグメントに解析される。セグメントのそれぞれは、入力データからの1つの新しい記号により付加された入力列の最初の部分の正確な複写から構成される。行われる複写はできるだけ長く、すでに解析されたセグメントに一致することに制約されない。出力セグメントを表す符

号語は、初めの複写部分が始まるところ、符号の長さ、および新しい記号へのポインタから成る情報を含む。レンペルージブのデータ圧縮技術についてのさらなる技術が、米国特許第4558302号に見られ、ここで参照により取り入れる。

【0011】前述のデータ圧縮手順は有効な一般用の無損失性手順であるが、冗長度のいくつかの特定の種類が、他の方法を使用して圧縮される。ランレングス符号化(run length encoding, RLE)として一般に知られるこのような無損失性方法の1つが、図形イメージデータに良く適している。RLEでは、一連の個々のキャラクタが、繰り返されたキャラクタの識別子を加えたカウント・フィールドとして符号化される。通常、2個のキャラクタが、それぞれのキャラクタの連なり(run)に印をつけるのに必要なので、この符号化は2またはそれより少ないキャラクタの連なりには使用されない。しかし、デジタルデータ形式で表される図形イメージを扱う場合には、任意の所与のラインにおいて同じキャラクタの大きな連なりが存在し、RLEはこのような情報について効率的な圧縮手順を行う。

【0012】前述のデータ圧縮手順のすべてが、データの冗長度に大きく依存し、顕著な圧縮比を達成する。これらの手順で明らかに不利なのは、データのある種類について、入力データが何らかの決まった冗長度に欠けているために入力よりも圧縮された出力が実際大きくなってしまうことである。印刷の技術分野では、このような「圧縮不完全な(incompressible)」データが簡単に生成される。画像のある種類は、「組織的ディザ(ordered dither)」または「誤差拡散(error diffused)」のどちらかとして分類される。組織的ディザの画像(「クラスタ(cluster)」ともいう)は、中間調(half-tone)画像であり、ページ全体にわたって中間調のグレー(gray)表示を含む。このような画像は、部分的なデータの冗長度を表すのが一般的であり、上に述べたようにデータ符号化の無損失性技術に向いている。

【0013】しかし、誤差拡散の画像(「分散(dispersed)」ともいう)は、それらのデータではほとんど冗長度を表さず、異なる圧縮方法が必要である。写真(photographic)画像を表す印刷データは、冗長度の低い印刷データの別の例を与える。結果として、ページプリンタの1つのデータ圧縮機構の使用は、もはやこのようなプリンタが画像データを取り扱うことをできなくしている。米国特許第5479587号のタイトル「Page Printer Having Adaptive Data Compression For Memory Minimization」はCambellらで提示され、ここで参照により取り入れる。これによると、ページプリンタは多様な圧縮技術を通じて歩んでおり、印刷されたデータの全ページに必要なサイズより少ない限られたメモリサイズに適応しようとしている。上述した出願では、メモリが少なくてイメージが印刷できない場合、第1の「モードM」圧縮技術

が使用される。この技術を使用して、各行に RLE を使用し、ブロック内の行から行に生じる変化デルタを符号化することにより、ブロックを圧縮しようという試みがなされる。「モードM」圧縮技術が、ページを印刷することができるので十分な圧縮比を提供するのに成功しない場合には、LZW型圧縮を使用して第2の試みがなされる。最後に、LZWベースの圧縮技術がページを印刷することができる十分大きな圧縮比を得るのに成功しない場合には、損失性圧縮手順が使用される。

【0014】

【発明が解決しようとする課題】ラスタ印刷データの処理において、印刷されるページを生成するのに先立ち、多様な操作がラスタ印刷データについて実行される。データ圧縮、カラー空間変換、中間調のような操作は、印刷されるページを生成するのに先だって実行することができる操作に含まれる。ラスタ印刷データの処理において、ページの多様なセクションが、異なる種類のデータ圧縮、カラー空間変換、中間調操作を使用して最適に処理される場合が頻繁にある。多様なラスタ印刷データ処理の操作をページの適切なセクションについて最適に実行することができるようになるため、ラスタ印刷データの最適処理に直面した良く起こる問題は、ページを形成するラスタ印刷データの分割であった。

【0015】例えば、ページに対応するラスタ印刷データを保管するのに必要なメモリ量を考えてみる。プリンタがドット場所の密度（インチあたりのドット）を大きくし、グレイスケール（gray scale）の機能を加え（グレイスケールのレベルを決めるためピクセルあたり多くのビット数を使用する）、カラー印刷の機能を含む（モノクロ印刷よりピクセルあたり追加のビットを必要とする）と、ページを印刷するのに使用するデータを保管するのに必要なメモリは、同じ解像度のモノクロ・プリンタに必要とされるメモリの32倍に達する。カラープリンタがより妥当なメモリサイズを使用することができるようになるため、データ圧縮技術がメモリの必要量を下げるのに通常使用される。しかし、ラスタ印刷データの異なる種類が、異なる圧縮技術を用いてそれぞれ最適に圧縮される。たとえば、イメージを含むページのセクションに対応するラスタ印刷データについては、圧縮比と印刷品質の最適な組み合わせは損失性圧縮技術を使用することにより達成される。しかし、テキストを含むページのセクションに対応するラスタ印刷データについては、圧縮比と印刷品質の最適な組み合わせは無損失性圧縮技術を使用することにより達成される。したがって、それぞれのデータ要素の処理にデータ処理操作の最適な型の適用を可能にするデータ処理パイプラインが必要である。

【0016】

【課題を解決するための手段】印刷データのバイトまたはワードのような複数のデータ要素から形成される複数

のデータストリームのそれぞれを別個に処理するためのデータ処理パイプラインは、各データ要素についてデータパイプライン処理操作の最適化を可能にする。データ処理パイプラインは、複数のデータストリームにより定められ、データ要素に対応する併合データ要素から形成される併合データストリームを使用する。たとえば、複数のデータストリームは、データ要素ごとにデータ要素のそれぞれの特性に基づいて印刷データストリームのようなソースデータストリームを分割することにより形成することができる。その特性に依存して、ソースデータストリームを2つより多くのデータストリームに最適に分割することができる。ラスタ印刷データを処理するのに使用されるデータ処理パイプラインの場合には、ラスタ印刷データのストリームをピクセルレベルで分割することができる。たとえば、ページの損失性領域に関連するピクセルは、損失性ラスタ印刷データストリームを形成することができ、ページの無損失性領域に関連するピクセルは、無損失性ラスタ印刷データストリームを形成することができる。その後、データ圧縮のようなデータ処理パイプラインの操作が、ページの各ピクセルについての圧縮比および印刷品質の最適な組み合わせを達成するよう選択される。

【0017】またラスタ印刷データのストリームは、ピクセルのそれぞれを表すのに使用されるビット数に基づいて分割することもできる。極端な場合、カラーで印刷されるページ領域に対応するピクセルは、ピクセルのそれぞれを表すのに24ビットを必要とすることができ、白黒で印刷されるページ領域に対応するピクセルは、ピクセルのそれぞれを表すのに1ビットのみを必要とすることができる。ピクセルのそれぞれを表すのに使用されるビット数に基づいてラスタ印刷データのストリームを分割することにより、ピクセルあたりのビット数について最適化された圧縮操作を使用することができる。さらに、バイナリの白と黒のデータのピクセルを含むラスタ印刷データのストリームは、カラー空間変換操作を必要としないのに対し、カラーデータのピクセルを含むラスタ印刷データのストリームはカラー空間変換操作を必要とすることができる。

【0018】さらに、ラスタ印刷データのストリームの分割は、ピクセルに対応するページ領域の所望の印刷解像度に基づくことができる。たとえば、ベクトルグラフィックス・オブジェクト(objects)のエッジ(edges)またはイメージ・オブジェクトのエッジは、通常印刷品質の理由により高解像度から良いものが得られ、ベクトルグラフィックス・オブジェクトの内部領域またはイメージ・オブジェクトの内部領域は、印刷品質を犠牲にすることなく低解像度で印刷することができる。解像度に基づいてラスタ印刷データストリームを分割することは、データ処理パイプラインの処理に必要なデータ量の減少を可能にし、これによりページを印刷するのに必要な時間

を減少させることができる。さらに、中間調および圧縮操作を、特定の分割されたラスタ印刷データストリームの解像度に基づいて、ラスタ印刷データの分割されたストリームのそれぞれについて最適化することができる。

【0019】データ処理パイプラインは、無損失性圧縮／伸張器、中間調ユニット、カラー空間変換器、またはそれらの組み合わせといった第1のパイプライン処理ユニットを含む。第1のパイプライン処理ユニットは、第1の入力を含み、複数のデータストリームのうちの1番目のものを受け取る。第1のパイプライン処理ユニットは、圧縮操作、伸張操作、カラー空間変換、中間調操作、または前述の操作の組み合わせといった第1の変換を、複数のデータストリームのうち1番目のデータストリームのデータ要素について実行するよう構成され、第1の変換されたデータストリームを生成する。

【0020】さらにデータ処理パイプラインは、損失性圧縮／伸張器、中間調ユニット、カラー空間変換器、またはそれらの組み合わせといった第2のパイプライン処理ユニットを含む。第2のパイプライン処理ユニットは、第2の入力を含み、複数のデータストリームのうちの2番目のデータストリームを受け取る。第2のパイプライン処理ユニットは、圧縮操作、伸張操作、カラー空間操作、中間調操作、または前述した操作の組み合わせといった第2の変換を、複数のデータストリームの2番目のデータストリームのデータ要素について実行するよう構成され、第2の変換されたデータストリームを生成する。

【0021】さらに、データ処理パイプラインは、第1の変換されたデータストリームおよび第2の変換されたデータストリームを受け取るよう構成された併合ユニットを含む。併合ユニットは、第1の変換されたデータストリームおよび第2の変換されたデータストリームを、併合データストリームを使用して出力データストリームへ併合するよう構成される。

【0022】

【発明の実施の形態】本発明は、ここで例示される典型的な実施の形態に限定されるものではない。さらに、印刷データ処理パイプラインの実施形態はカラーレーザー・プリンタについて述べるが、当該技術分野の当業者には、この開示内容を理解した後で、無損失性または損失性データ圧縮方法を用いて選択的に圧縮されたデータを取り扱う並列バスを使用することにより、パフォーマンスを有利に達成するため、開示された印刷データ処理パイプラインの構成が他の画像システムにも適用することができることがわかるであろう。たとえば、開示された印刷データ処理パイプラインは、スキャナおよびデジタル写真処理のラボ(laboratories)に通常適用することができる。さらに、画像データの保管にメモリを必要とする画像を生成するための任意のシステムが、最適化された画像品質を達成することができ、開示されたパイプラ

インの構成を適用して必要なメモリ量を最小とすることができる。

【0023】カラー印刷では、印刷品質を高レベルに維持する一方、印刷データを保管するのに必要なメモリを下げる印刷データ処理パイプラインが必要である。損失性または無損失性の圧縮方法を使用して選択的に印刷データを圧縮する印刷データ処理パイプラインの構成を使用することは、高い印刷品質を維持し、印刷データの保管に必要なメモリを最小化する。さらに、印刷データ処理パイプラインにおける操作を適切な位置におくことにより、印刷品質をさらに良くし、必要なメモリ量を減少させる。

【0024】図1は、複数のデータ要素から形成されたソースデータストリームの区画から生成された複数のデータストリームを処理するためのデータ処理パイプライン200の一般的な実現を表すブロック図である。複数のデータストリームのそれぞれは、データ要素から形成される。図1には、複数のデータストリームのうちの第1のデータストリーム201および第2のデータストリーム202のみしか表していないが、データ要素の特性に依存して、ソースデータストリームは2より多くのデータストリームに分割して、データ要素を最適に処理することができる。

【0025】第1のパイプライン処理ユニット203は、複数のデータストリームのうち第1のデータストリーム201について第1の変換を実行するのに使用される。第1のパイプライン処理ユニット203により実行される第1の変換は、データ圧縮または伸張、カラー空間変換、中間調処理、または他のデータ処理操作のような1または複数の操作を行うことができる。第1の変換を実行する方法は、複数のデータストリームのうちの第1のデータストリーム201を形成するデータ要素の特性に最適化される。第1のパイプライン処理ユニット203は、第1の変換されたデータストリーム205を生成する。

【0026】第2のパイプライン処理ユニット204は、複数のデータストリームのうち第2のデータストリーム202について第2の変換を実行するのに使用される。第2のパイプライン処理ユニット204により実行される第2の変換は、データ圧縮または伸張、カラー空間変換、中間調処理、または他のデータ処理操作のような1または複数の操作を行うことができる。第2の変換を実行する方法は、複数のデータストリームのうちの第2のデータストリーム202を形成するデータ要素の特性に最適化される。第2のパイプライン処理ユニット204は、第2の変換されたデータストリーム206を生成する。

【0027】併合ユニット207は、第1の変換されたデータストリーム205および第2の変換されたデータストリーム206を受け取る。併合ユニット207は、

第1の変換されたデータストリーム205および第2の変換されたデータストリーム206を、出力データストリーム208に併合する。この併合操作を行うため、併合ユニット207は併合データストリーム209を使用する。併合データストリーム209は、併合データ要素から形成され、複数のデータストリームのうちの第1のデータストリーム201および第2のデータストリーム202により定められるため、併合操作の結果である出力データストリーム208のデータ要素は、ソースデータストリームにおけるデータ要素の順番を有する。よって、併合ユニット207は、第1の変換されたデータストリーム205および第2の変換されたデータストリーム206を組み立て、出力データストリーム208にソースデータストリームにおけるデータ要素の順番を保持する。

【0028】ソースデータストリームにおけるデータ要素の特性に基づいて、ソースデータストリームを分割して複数のデータストリームのうちの第1の201および第2の202を形成することにより、第1のパイプライン処理ユニット203および第2のパイプライン処理ユニット204は、データ要素の処理を最適化するようそれぞれ構成される。

【0029】図2は、カラープリンタを通る印刷データの流れを表す簡略化された概念図であり、カラーレーザ・プリンタについての印刷データ処理パイプライン15の好ましい実施形態を含む。図2は、並列の損失性および無損失性のパスを使用し、印刷データ処理パイプライン15により実行される典型的な操作を通る印刷データの流れのみの例示を試みたものである。図2は、印刷データ処理パイプライン15のハードウェア・ブロック図ではない。ハードウェア・ブロック図の後の説明から明らかになるように、印刷データ処理パイプライン15は、印刷データの処理を最適化するよう様々な異なる方法で構成することができる。

【0030】システムへの印刷データ入力は、いくつかの異なる形式で送られてくる。入力印刷データは、表示リスト(Display List)、ラスタ印刷データ、または既に圧縮を受けたラスタ印刷データの形で送られてくる。表示リスト印刷データは、印刷されるべきページを構成するのに必要な情報を含む。表示リスト印刷データは、テキストまたはイメージ印刷データとしてのラスタ印刷データ、テキスト文字を表す印刷制御言語のコード、グラフィックス・イメージを表すグラフィックス言語のコード、またはこれらの印刷データのいくつかの組み合わせを識別するコードと一緒に、ラスタ印刷データを含むことができる。概念的には、印刷データは、印刷出力を生成するのに必要な形式に印刷データを変換するのに必要とされる処理に依存した適切な場所で、印刷データ処理パイプライン15に入力される。

【0031】画像処理は、ファームウェア・ルーチンの

実行により画像処理操作1において、表示リスト印刷データについて実行される。この画像処理操作1には、入力をページ片(page strips)に分割することが含まれる。入力をページ片に分割するため、印刷データが対応するページの垂直位置に基づいて、表示リスト印刷データは保管される。画像処理操作1でページ片にページを分割することには、形成されるページ片に対応するページのセクションを形成する印刷データの種類を考慮することを伴う。各ページ片は、1または2のページ片要素で形成される。ページのセクションが、完全に損失性または完全に無損失性の印刷データのどちらかから形成されると、1要素のページ片のみが、ページのセクションについて画像処理操作1で形成される。しかし、そのセクションについての表示リスト印刷データが損失性および無損失性の印刷データの両方を含むときは、2つの別個のページ片要素が、そのページのセクションについて形成される。これらのページ片要素の一方は損失性印刷データのみを含み、他方は無損失性印刷データのみを含む。

【0032】対応するページのセクションについての損失性および無損失性のページ片要素の形成は、データの両種類が表示リスト印刷データに存在する時に発生する。この場合、画像処理操作1は、損失性および無損失性の両方のページ片要素をピクセル位置ごと適切に、ページ・セクションからの損失性および無損失性のページ片要素で満たすことにより、2つのページ片要素のそれを作成する。損失性および無損失性のページ片要素には、ページのセクションについての印刷データが含まれる。たとえば、表示リスト印刷データが、損失性イメージ印刷データと一緒に無損失性テキスト印刷データを含む場合には、画像処理操作1は表示リスト印刷データをテキスト印刷データとイメージ印刷データとに分離する。その後、無損失性および損失性のページ片要素のそれぞれが、ピクセル位置ごとに、それぞれのテキスト印刷データおよびイメージ印刷データで満たされる。この場合、表示リスト印刷データに対応するページ片は、無損失性テキスト印刷データを包含するページ片要素で被せられる損失性イメージ印刷データを包含するページ片要素、を含む合成ページ片とみなされる。

【0033】損失性および無損失性のページ片要素が形成される場合には、画像処理操作1が対応する併合プレーン(merge plane)を生成する。この併合プレーンは、2つのページ片要素において対応するピクセル位置のそれぞれの対について1ビットから成り、そのピクセルについての印刷データが含まれる損失性および無損失性のページ片要素を指す。併合プレーンを形成するビットは、バイトに形づくられ、バイト・ストリームとして印刷データ処理パイプライン15を通って転送される。この併合プレーンは、印刷データ処理パイプライン15の後の段階で使用され、無損失性および損失性のページ片

要素を組み合わせ、これにより最初の（オリジナルの）ページ片が再構成される。この併合操作は、この明細書の後の方で詳細に述べる。

【0034】1つのページ片要素が、損失性印刷データのみ、または無損失性印刷データのみを有するよう形成される場合は、そのページ片についてのすべての印刷データが1つのページ片要素に含まれる。さらに、ページ片要素に関する併合プレーンは、イメージ処理操作1によっては生成されない。ページのセクションについて1つのページ片要素しかないので、併合プレーンのどのビットも同一である。したがって、印刷データ処理パイプライン15を通るすべて同じ値のバイトから成る併合プレーンを送る必要性が無い。イメージ処理操作1で併合プレーンを生成する代わりに、印刷データ処理パイプライン15の後の段階で併合プレーンが生成される。これについては、この明細書の後の方で詳しく述べる。

【0035】また画像処理操作1は、ページ片における各ピクセル位置に対応する1つの値を有する中間調プレーン(halftone plane)を生成する。この明細書で後で述べるように、各中間調の値は2ビットで表され、ピクセルに対応する印刷データに適用される中間調操作を決定する。実行される中間調操作が、ページ片の各ピクセルについて同じである場合は、中間調プレーンは印刷データ処理パイプライン15の後の方の段階で生成される。これについては、この明細書の後の方でより詳細に述べる。

【0036】ページ片要素に分割された表示リスト印刷データから、ラスタ印刷データが表示リスト印刷データに対応して生成される。結果のラスタ印刷データのページ片要素は、生成される時に印刷データ処理パイプライン15の操作を通る。ラスタ印刷データメモリ2は、画像処理操作1により生成されたラスタ印刷データを保管するのに使用される。ラスタ印刷データメモリ2に割り振られるメモリ領域は、ページ片あたりの最大限可能なライン数をもつ2つのページ片要素のラスタ印刷データ、対応する併合プレーンのサイズ、対応する中間調プレーンのサイズを含むのに十分な記憶容量をもつ。ページ片要素のそれぞれは、画像処理操作1においてラスタ処理され、ラスタ印刷データメモリ2に割り振られたメモリ領域に保管され、その後印刷データ処理パイプライン15の次の操作に送られ、画像処理操作1でラスタ処理された次のページ片要素のためにラスタ印刷データメモリ2に領域を作る。好ましい実施形態においては、ラスタ印刷データメモリ2が、印刷データ処理パイプライン15が実装される集積回路に含まれないことに注意すべきである。

【0037】通常、生成されたラスタ印刷データは、それぞれのピクセルについて3個の8ビットのバイトから成る。3バイトのそれぞれが、表示リスト印刷データが表現されるカラー空間におけるカラーディメンション(color dimension)

の1つに対応する。印刷データ処理パイプラインの好ましい実施の形態が動作するカラープリンタでは、これはRGBカラー空間である。しかし、当該技術分野の当業者には、入力印刷データのカラー空間が、シアン、マゼンタ、黄、黒(CMYK)や、色相(hue)、飽和度(saturation)、明度(value)(HSV)、色相、明度(lightness)、飽和度(HLS)、輝度(luminance)、赤黄スケール、緑青スケール(L*a*b*)、輝度、赤緑スケール、黄青スケール(Luv)、輝度、赤青スケール、緑黄スケール(YCrCb)またはYIQのような任意のカラー空間であることがわかるであろう。生成されたラスタ印刷データは、イメージ、テキスト、線画、グラフィックス、またはこれらのいくつかの組み合わせに対応する。

【0038】前述したように、画像処理操作1は、無損失性および損失性のページ片要素が形成されるような表示リスト印刷データの場合に、併合プレーンを作成する。印刷データ処理パイプラインの後の方で、併合ビットは対応する無損失性および損失性のページ片要素の対からピクセルを選択するのに使用され、無損失性および損失性のページ片要素を組み合わせ、最初の（オリジナルの）画像に向けてピクセル間の正確な空間的関係を維持する。

【0039】カラー空間変換は、カラー空間変換操作3で実行される。ラスタ印刷データについて達成される圧縮の度合いは、ラスタ印刷データが表現されるカラー空間により影響される。たとえば、高い損失性圧縮比（同じ画像品質で）は、RGBカラー空間からYCrCbカラー空間へのカラー空間変換を実行し、YCrCbへの変換後に損失性圧縮操作を実行することにより成し遂げられるのが通常である。YCrCbカラー空間は、輝度/色相/彩度(chroma)型のカラー空間であり、カラー空間のCrおよびCb要素は、それぞれ色相および彩度の両方の情報を含む。人間の眼は、輝度の変更に対して最も敏感であり、彩度の変更に対しては相対的に敏感ではない。このため、このカラー空間で表現される印刷データは、CrおよびCb要素において著しい冗長度の量を含む。結果として損失性圧縮は、相対的に大きい圧縮比でラスタ印刷データの損失性ページ片要素について実行される。圧縮比に依存したこのカラー空間の有利な点は、無損失性圧縮については存在しない。その理由は、無損失性ページ片要素は、RGBカラー空間からYCrCbカラー空間へのカラー空間変換を受けないからである。しかし、このカラー空間変換が他の理由で必要とされた場合には、実行される。印刷データ処理パイプラインの後の段階で、無損失性ページ片要素および損失性ページ片要素の両方が、RGBまたはYCrCbカラー空間からCMYKカラー仕様へのカラー空間変換を受けれる。このカラー空間変換は、この明細書の後の方で述べる。

【0040】1つのページ片に対応するページのセクションから無損失性および損失性のページ片を形成し、さらに併合プレーンおよび中間調プレーンを生成することは、印刷データ処理パイプライン15を通って送られるデータ量を増加させる。それぞれのピクセルに関連する併合ビットおよび2つの中間調ビットが、印刷データ量を増加させる。しかし、高圧縮比が無損失性および損失性のページ片について行われ、しかも併合プレーンおよび中間調プレーンも無損失性圧縮を受けるので、印刷データを包含するのに必要なメモリ領域は、最悪の場合でわずかに増加するだけである。しかし、無損失性および損失性のデータに異なるデータ圧縮のアルゴリズムを適用しないパイプラインを介した印刷品質の改良結果は、顕著である。ほとんどの印刷ジョブにとって、印刷データはすべて損失性か、またはすべて無損失性かのどちらかである。これらの場合、画像処理操作1から印刷データ処理パイプライン15を通って送られるラスタ印刷データの量における唯一の増加は、ページ片のピクセルに異なる中間調アルゴリズムを適用する場合には中間調プレーンである。同じ中間調アルゴリズムがページ片のすべてのピクセルに適用される場合には、イメージ処理操作1では中間調プレーンはまったく生成されない。

【0041】ラスタ印刷データは、印刷データ処理パイプラインに直接入力することができる。ラスタ印刷データは、スキャナまたはホスト・コンピュータから送られることができる。またラスタ印刷データを提供する装置は、画像処理操作1により使用されるヘッダを提供し、ページ片を形成する。このヘッダには、画像処理操作1のため損失性または無損失性としてラスタ印刷データを分離するのに画像処理操作1が使用する情報が含まれる。

【0042】損失性ラスタ印刷データのページ片要素が損失性圧縮を受ける前に、ラスタ印刷データのカラー空間変換されたページ片要素のCrおよびCb要素が選択的に減少させられ、印刷データ処理パイプライン15が取り扱わなければならない印刷データの量を減少させ、これによりデータ処理量を増加させ、ラスタ印刷データを保持するのに必要なメモリ量を減少させる。前述したCrおよびCb要素の冗長度が、このラスタ印刷データのサブサンプリングが達成されることを可能にし、ラスタ印刷データの視覚的な無損失性の特性を維持する。サブサンプリング操作は、損失性圧縮操作5で実行される。損失性圧縮操作5は、走査線に沿って8ピクセル幅であって、8の連続した走査線分の高さであるセクションから形成されるラスタ印刷データのブロックについて圧縮を行う。サブサンプリングを通り、これらの8ピクセルx8ピクセルのブロックの全体数が減少する。

【0043】発生するサブサンプリングの量は必要なデータ減少量に依存し、ラスタ印刷データが使用可能なメモリ領域に適合することができるようとする。十分なメ

モリ領域が使用可能であれば、サブサンプリングはページ片要素の損失性ラスタ印刷データの量を1/3だけ下げるよう実行される。メモリへの損失性ラスタ印刷データの適合が必要な場合には、サブサンプリングはページ片要素の損失性ラスタ印刷データの量を半分だけ下げるよう実行される。

【0044】1/3の減少を達成するサブサンプリングは、以下のように行われる。たとえば、損失性ラスタ印刷データの3つのグループを考え、それぞれのグループが1つのカラー空間からピクセルの8x8の4つのブロックを表すとする。第1グループは輝度要素から形成され、第2グループはCr要素から形成され、第3グループはCb要素から形成される。そして、損失性ラスタ印刷データが全体として12ブロックある。第2グループでは、4ブロックそれぞれにおける8の走査線のそれぞれについて、各走査線に沿った交互のピクセルに対応する損失性ラスタ印刷データのみが維持される（各ブロックの各走査線の第1番目のピクセルで始まる）。結果として、4ブロックそれぞれの各走査線について、8ピクセル幅が4ピクセルに減少する。この方法では、Cr要素の4ブロックは、2ブロックにサブサンプリングされる。Cb要素のサブサンプリングは、同様のやり方で、4ブロックから2ブロックになる。このサブサンプリングの後、ピクセルの最初の12ブロックのうち8個はそのままあり、すなわち1/3だけ減少する。

【0045】半分の減少を達成するサブサンプリングは、Cr要素の2ブロックおよびCb要素の2ブロックが残る1/3の減少を行うのと同じサブサンプリング手順を適用することによりなされる。同じサブサンプリング手順を、CrおよびCb要素の残りのブロックに適用すると、Cr要素について1ブロック、Cb要素について1ブロックという結果を生じる。サブサンプリングのこの第2のパスの後、ピクセルの最初の12ブロックのうち、6が残り、半分だけ減少する。

【0046】サブサンプリング操作が実行された後、損失性圧縮操作5において、損失性圧縮アルゴリズムがページ片要素に適用される。印刷データ処理パイプライン15の好ましい実施形態において使用される損失性圧縮方法は、周知のJPEGアルゴリズムである。JPEG損失性圧縮方法が選択されたのは、以前開発されたハードウェアが、この方法を実行するのに簡単に利用することができたからである。しかし、ベクトル量子化のような任意の損失性圧縮方法が使用してきた。無損失性ページ片要素である場合には、このページ片要素の圧縮は、無損失性圧縮操作4で実行される。さらに、ページ片要素に対応する併合および中間調プレーンがある場合には、これらの両方が無損失性圧縮操作4で圧縮される。印刷データ処理パイプライン15の好ましい実施形態では、ジブーレンペルの無損失性圧縮／伸張方法を使用し、これは米国特許第5455576号で開示され、ここで

参照により取り入れる。しかし、JBIG、ランレンジスまたはデルタロウ圧縮(delta row compression)のような任意の無損失性圧縮／伸張方法も使用することができる。無損失性および損失性圧縮を実施するのに使用する技術に関する情報は、「INTRODUCTION TO DATA COMPRESSION」、Khalid Sayood、1996年、Morgan Kaufmann出版という本に見られ、ここで参照により取り入れる。

【0047】損失性圧縮操作5および無損失性圧縮操作4によりそれぞれ生成された損失性および無損失性の圧縮ラスタ印刷データは、圧縮ラスタ印刷データメモリ6に保管される。圧縮ラスタ印刷データメモリ6に割り振られる領域は、ページ全体の圧縮損失性および無損失性のラスタ印刷データの3つの色のプレーンすべて、ページ全体の圧縮中間調データ、ページ全体の圧縮併合データを保持するのに十分なサイズである。好ましい実施形態においては、圧縮ラスタ印刷データメモリ6が、印刷データ処理パイプライン15が実装される集積回路に含まれることに注意すべきである。さらに、ラスタ印刷データメモリ2および圧縮ラスタ印刷データメモリ6の両方が、圧縮ラスタ印刷データであろうと、圧縮／伸張操作が実行されなかったラスタ印刷データについてであろうと、物理的にシステムメモリに置かれ、印刷データを保管する。メモリは所与の時間で保管されるデータの種類によって指定しさえすればいいので、システムメモリの場所は、異なる時間の印刷データの両方の種類を保管するのに使用することができる。

【0048】損失性または無損失性のページ片要素の各ピクセルについてラスタ印刷データは3バイトで表され、すなわちラスタ印刷データのそれぞれのカラー空間の要素について1バイトで表される。損失性および無損失性のページ片要素は、印刷データ処理パイプライン15を通じて2つの並列チャネルを移動する。印刷データ処理パイプライン15における印刷データの位置により、損失性および無損失性のラスタ印刷データストリームは、1バイト幅または3バイト幅のどちらかである。損失性および無損失性のラスタ印刷データは、カラー空間変換操作3、9に3バイト幅ストリームとして入り、1バイト幅ストリームとして出て、出力カラー空間の次元を表す3バイトグループの各バイトを有する3バイト幅ストリームにパックされる。損失性および無損失性のラスタ印刷データは、カラー空間変換操作12に3バイト幅ストリームとして入り、1バイト幅ストリームとして出る。カラー空間変換操作3、9、12に入る3バイトグループのそれぞれのバイトが、入力カラー空間の1次元に対応するラスタ印刷データを表す。カラー空間変換操作12を出る結果としての1バイトは、印刷を受けている出力カラー空間の1つの次元に対応するラスタ印

刷データを表す。損失性および無損失性の印刷データは、画像処理操作1を3バイト幅ストリームとして出る。損失性および無損失性の印刷データが、損失性5、8、11圧縮および無損失性4、7、10圧縮操作に3バイト幅ストリームとして入り、1バイト幅ストリームとして出る、その後バッファされ、3バイト幅ストリームとなる。損失性および無損失性の印刷データは損失性5、8、11伸張および無損失性4、7、10伸張操作に1バイト幅ストリームとして入り、1バイト幅ストリームとして出る。損失性および無損失性のラスタ印刷データは、併合操作13および中間調操作14の両方を1バイト幅ストリームとして入り、1バイト幅ストリームとして出る。

【0049】併合データプレーンはバイトに調整される。各ピクセルについて1ピットのみが併合操作に必要とされるので、併合データの各バイトは、8ピクセルについての併合情報を含む。中間調データプレーンは8ビットのバイトに調整される。各ピクセルについて2ビットが中間調操作では必要とされるので、中間調データの各バイトは4ピクセルについての中間調情報を含む。中間調プレーンおよび併合プレーンの両方が、印刷データ処理パイプラインを通じて2つの並列バイト幅チャネルを移動する。

【0050】損失性および無損失性のページ片要素を形成し、その後これらのページ片要素のラスタ印刷データを、それぞれ損失性または無損失性の圧縮方法を用いて圧縮することにより、メモリ圧縮および画像品質の最適な組み合わせが印刷データについて達成される。損失性ラスタ印刷データは損失性圧縮技術を用いて高圧縮され、視覚的に無損失性の特性を維持する。無損失性ラスタ印刷データは無損失性圧縮技術を用いて圧縮されるので、画像品質の劣化を生じず、高圧縮比を達成する。選択された損失性および無損失性の圧縮方法は、損失性ラスタ印刷データ（イメージ）および無損失性ラスタ印刷データ（テキスト、線画、グラフィックス）について最適化され、画像品質を劣化させることなく高圧縮比を生じさせる。この機能は非常に有利な点であり、印刷データ処理パイプライン15において損失性および無損失性のページ片要素を使用することから生じる。印刷データ処理パイプライン15の好ましい実施形態では、複数のチャネルを使用し、損失性および無損失性のラスタ印刷データ、対応する中間調および併合データを、印刷データ処理パイプライン15を通じて移動させるが、1つのマルチプレクサ・チャネルを使用して、連続的に印刷データのページ片要素を転送することもできることに注意すべきである。1つのマルチプレクサ・チャネルの使用は、必要なハードウェアを縮小するためにパフォーマンスを犠牲にする。

【0051】圧縮ラスタ印刷データを印刷データ処理パイプライン15に入力することが可能である。これは、

圧縮ラスタ印刷データが表示リストのコマンドに関連する印刷データ処理パイプライン15の入力である時に生じることができ、ラスタ印刷データまたは圧縮ラスタ印刷データがページ全体の伸張および印刷に備えて印刷データ処理パイプラインの入力であることができる。この圧縮ラスタ印刷データは、ホストコンピュータまたはホストコンピュータを介したスキャナにより与えることができる。この圧縮ラスタ印刷データには、画像処理操作1により使用される情報を含むホストコンピュータにより付されるヘッダが含まれる。

【0052】表示リスト印刷データとしての印刷データ処理パイプライン15への損失性イメージの前の入力を、圧縮損失性ラスタ印刷データとして印刷データ処理パイプライン15への入力である損失性イメージで被せたい(overlay)と望む場合を考えてみる。この場合、印刷データ処理パイプライン15にパスが提供され、圧縮ラスタ印刷データを圧縮ラスタ印刷データメモリ6から損失性伸張操作8へ送り、カラー空間変換操作9でカラー空間変換を行って(圧縮損失性ラスタ印刷データがRGBカラー空間で表現されない場合について)、RGBカラー空間に戻し、結果をラスタ印刷データメモリ2に保管し、前の入力損失性イメージを用いて被せることを行うため、伸張損失性ラスタ印刷データを画像処理操作1に配信する。その後画像処理操作1は、損失性ページ片要素への分割を実行する。圧縮損失性ラスタ印刷データを無損失性印刷データと組み合わせるためのこのパスが、概念的意味で図2に示されている。印刷データがハードウェアの機能ブロックに追従し、図2に示される操作を行う実際のルートは、この明細書の後ろの方で述べる。

【0053】前述したように、印刷データ処理パイプライン15は、ラスタまたは圧縮ラスタ印刷データ形式の印刷を供給することができるホストコンピュータまたはスキャナのような多様な装置から、ラスタまたは圧縮ラスタ印刷データを受け取ることができる。ラスタまたは圧縮ラスタ印刷データのどちらかを受け取ることは、ラスタ操作を行う画像処理操作1の必要性を取り除く。さらに、印刷データが、もう印刷ができる圧縮ラスタ印刷データとして受取られると、損失性5および(または)無損失性4圧縮操作またはカラー空間変換操作3を行う必要性がない。

【0054】印刷データ処理パイプライン15の好ましい実施形態に示される操作を達成するのに使用される機能ブロックは、1つの集積回路に実現される。デジタル集積回路でこれらの機能を実現するのに使用される技術は、デジタル集積回路の設計分野において周知である。専用ハードウェアにおける印刷データ処理パイプライン15の機能の実現は、パイプラインに入る印刷データについてすぐれた処理パフォーマンスを提供する。

【0055】マイクロプロセッサを使用して印刷データ

処理パイプライン15における操作のそれぞれを実行することは可能であるが、印刷データの処理パフォーマンスが極端に低下する。マイクロプロセッサは、必要な印刷データ圧縮、カラー空間変換、印刷データ伸張操作を実行するのに、損失性および無損失性のページ片要素を交互に取り扱う必要がある。さらに、マイクロプロセッサは、損失性および無損失性のページ片要素を併合し、中間調操作を実行する。

【0056】印刷データ処理パイプライン15の好ましい実施形態は、CMYKカラー仕様で表現される印刷データを受け取るよう設計されるカラーレーザ・プリントエンジン16を用いて動作する。さらに、プリントエンジン16は、トナーをページに転送するのに先立ち、CMYKカラー仕様のそれぞれのプレーンについて連続的にトナーを現像(developing)することにより動作する。プリントエンジン16によって使用される4パスの現像処理が、印刷データ処理パイプライン15を通るラスタ印刷データが流れる方法を命令する。レーザプリントエンジン16のこの動作モードにより、ページについてのラスタ印刷データが、4つの連続したパスでプリントエンジン16に送られる(CMYKカラー仕様のそれぞれの次元について1パス)。

【0057】前述したように、ページ全体の圧縮ラスタ印刷データは、損失性および無損失性のページ片要素が圧縮される時に、圧縮ラスタ印刷データメモリ6に割り振られたメモリに累積される。損失性圧縮ラスタ印刷データはYCbCrカラー空間で表現される。無損失性圧縮ラスタ印刷データは、RGBカラー空間で表現される。レーザプリントエンジンのCMYKカラー仕様の4つのプレーンのそれぞれに圧縮ラスタ印刷データを変換するには、4つの連続したパスで印刷データ処理パイプライン15を通って、ページについての圧縮ラスタ印刷データを送る必要がある。これら4パスのそれぞれにおいて、無損失性および損失性の圧縮ラスタ印刷データが、2つの1バイト幅チャネルを介してそれぞれの無損失性10および損失性11伸張操作に、伸張のため送られる。無損失性圧縮ラスタ印刷データのバイトにインターリーブ(interleave、はさみこみ)されるのは、圧縮併合プレーンおよび中間調プレーンである。併合プレーンおよび中間調プレーンが無損失性ラスタ印刷データにインターリーブされるので、これらは無損失性ラスタ印刷データと一緒に無損失性圧縮を受ける。伸張の後、併合プレーンおよび中間調プレーンは、伸張無損失性印刷データストリームから抽出される。これについては、この明細書の後の方で詳しく述べる。損失性圧縮印刷データは、損失性ページ片要素を含む。損失性圧縮ラスタ印刷データの伸張の後、補間操作が実行され、損失性ラスタ印刷データの圧縮に先立ちサブサンプリングを通して除去されたCrおよびCb要素の値を拡張(expand)する。補間操作は、損失性伸張操作11の一部として実行

される。

【0058】損失性 11 および無損失性 10 の伸張操作の後、無損失性印刷データおよび損失性印刷データの両方が、カラー空間変換操作 12 のカラー空間変換のために送られる。プリントエンジン 16 が 4 パスの現像処理を使用するので、ページのすべてのカラープレーン (color plane) についての圧縮ラスタ印刷データが、損失性 11 および無損失性 10 伸張操作を介して 4 回連続して送られる。無損失性 10 および損失性 11 の伸張操作を通る圧縮ラスタ印刷データのカラー空間変換操作 12 への 4 パスのそれぞれについて、カラー空間変換操作 12 が CMYK カラー仕様の 4 プレーンのうち 1 つを生成する。カラー空間変換操作 12 は、印刷されるべきページのピクセルのそれぞれを表す損失性または無損失性のラスタ印刷データの 3 個の 8 ビットのバイトを使用して実行する。カラー空間変換操作 12 からのラスタ印刷データストリーム出力は、損失性ラスタ印刷データおよび無損失性ラスタ印刷データから形成される 2 個の 8 ビットチャネルを含む。カラー空間変換操作 12 への無損失性ラスタ印刷データ入力は、RGB カラー空間で表現される。カラー空間変換操作 12 への損失性ラスタ印刷データ入力は、YCrCb カラー空間で表現される。

【0059】図 2 は、損失性 5 および無損失性 4 圧縮操作、無損失性 7 および損失性 8 伸張操作、無損失性 10 および損失性 11 伸張操作が、別個のハードウェアユニットで達成されることを示唆する。しかし、印刷データ処理パイプライン 15 の好ましい実施形態においては、損失性の圧縮および伸張を実行する 1 つのハードウェアユニットがあり、無損失性の圧縮および伸張を実行する 1 つのハードウェアユニットがある。これらのユニットは、印刷データ処理パイプライン 15 についてすべての圧縮／伸張操作を行う。印刷データ処理パイプライン 15 を説明する目的のために、好ましい実施形態において、図 2 は別個のハードウェアユニットとしてカラー空間変換操作 3, 9, 12 を示すが、1 つのカラー空間変換器が、必要なカラー空間変換のすべてを実行するのに使用される。印刷データ処理パイプライン 15 の好ましい実施形態においては、カラー空間変換器が、出願「APPARATUS FOR GENERATING INTERPOLATION INPUT DATA」(10960608-1)、「APPRATUS FORROUTING INTERPOLATION INPUT DATA」(10960467-1) に開示されており、参照によりここで取り入れる。当該技術分野の当業者には、損失性および無損失性の印刷データを表現するのに選択されるカラー空間の間で変換を実行するカラー空間変換器、および複数パス印刷データ処理パイプラインで使用するのに適したレーザプリントエンジンにより使用されるカラー仕様を、この明細書を理解した後でわかるであろう。

【0060】併合操作 13 は、無損失性ラスタ印刷データストリーム、損失性ラスタ印刷データストリーム、中間調データストリーム、併合データストリームを入力と

して受け取る。これらの印刷データストリームのそれぞれが、バイトストリームから形成される。ページ全体のピクセルのそれぞれについてプリントエンジンのカラー仕様の 1 つのカラープレーンを表す印刷データが、無損失性および損失性のラスタ印刷データストリームに含まれる。併合操作 13 の機能は、ピクセルごとにこれら 2 つのラスタ印刷データストリームを組み合わせ、前に無損失性および損失性のページ片要素に分割された最初の (オリジナルの) 画像を、処理されているカラープレーンについて再構成する。各ピクセルに関連する併合データのビットが併合操作 13 により使用され、現在処理されているピクセルのラスタ印刷データが、無損失性ラスタ印刷データストリームから選択されるか、または損失性ラスタ印刷データストリームから選択されるかどうかを判断する。併合操作 13 を通る損失性および無損失性のラスタ印刷データの引き続き起こる次のパスが、プリントエンジンのカラー仕様の残っているカラープレーンについて最初の画像を再構成する。ピクセルのそれぞれが、印刷されたページ上の位置に対応する。再構成処理は、無損失性および損失性のラスタ印刷データストリームの空間的同期とみなされ、最初の画像を生成する。さらに、空間的同期を達成するため、無損失性および損失性のラスタ印刷データストリームと併合データのストリームは、時間的に同期しなければならない。

【0061】開示された印刷データ処理パイプライン 15 は、プレーンごとにプリントエンジン 16 の現像特性を利用し、印刷操作をするのに必要なメモリ全体量を下げる。印刷データ処理パイプライン 15 において、カラー空間変換操作 12 の次に併合操作 13 を位置づけることで、プリントエンジン 16 の CMYK カラー仕様の 1 つのカラープレーンについてラスタ印刷データの保管のため、併合操作 13 が最小サイズのラインバッファを使用するという必要性のみがある。併合操作 13 がカラー空間変換操作 12 に先だって印刷データ処理パイプライン 15 に位置したのでは、RGB および YCrCb カラー空間の 3 次元のそれぞれについてラスタ印刷データを保管するのに十分なサイズのラインバッファを併合操作 13 が使用するという必要性が生じることとなる。

【0062】無損失性および損失性印刷データについて別個のパスを使用する印刷データ処理パイプラインの構成は、複数パスの代わりに 1 つのパスでプリントエンジンのカラー仕様のすべてのカラープレーンについてラスタ印刷データを受けることにより動作するプリントエンジンと互換性がある。この場合、カラー空間変換操作 12 が、各ピクセルを表すラスタ印刷データについてカラー空間変換を実行し、同時に C、M、Y、K 要素の対応する値のそれぞれを生成する必要がある。これは、C、M、Y、K 要素それぞれを生成するのに必要な補間を同時に実行する機能を有するカラー空間変換操作 12 を必要とする。さらに併合操作 13 は、C、M、Y、K 要素

それぞれについてラスタ印刷データを含むのに十分なサイズのラインバッファが必要である。

【0063】印刷データ処理パイプライン15の好ましい実施形態では、併合操作13で組み合わされたラスタ印刷データは、中間調操作14に付される。中間調操作14は、プリントエンジン16への印刷データの配信に先だってラスタ印刷データについて実行される最後の操作となるよう位置づけられる。中間調操作14は、各ピクセルに関連する中間調印刷データによって示されたように、併合されたラスタ印刷データについてプレーンごとに中間調処理を行う。併合操作13の後に中間調操作14を位置づけることが、中間調操作4がプリントエンジン16により使用されるカラー空間の各カラープレーンについて連続的に処理することを可能にし、これによってラスタ印刷データを保管するのがより少ないメモリで足りる。さらに、中間調操作14が併合操作に先だって実行されると、無損失性ラスタ印刷データおよび損失性ラスタ印刷データのストリームについて中間調操作14を別個に実行する必要性があるために、中間調操作14の複雑さが著しく増加する。さらに、印刷データ処理パイプライン15の最後に中間調操作14を置くことが、最適な印刷品質をもたらす。損失性ラスタ印刷データの圧縮の事前に中間調操作14を位置づけることは、中間調処理により達成された強調(enhancements、エンハンスメント)のいくらかの損失をもたらし、中間調ラスタ印刷データについて達成された圧縮レベルを減少させる。加えて、印刷品質の最高レベルが、プリントエンジンのカラー仕様で表されるラスタ印刷データについて中間調操作14を実行することにより達成される。したがって、印刷データ処理パイプライン15の好ましい実施形態では、中間調操作14はカラー空間変換操作12の後に位置される。

【0064】ラスタ印刷データの各ピクセルに中間調ビットを割り当てるのは、中間調プレーンを作る画像処理操作1で実行される。前述したように、印刷データ処理パイプラインに入る印刷データは、ラスタ印刷データ形式のテキスト、グラフィックス、イメージ印刷データ、印刷制御言語の印刷データ、グラフィックス言語の印刷データであることができる。画像処理ブロック操作1のファームウェア操作は、適用される中間調アルゴリズムの種類に従って対応する損失性および無損失性のページ片要素について中間調プレーンを生成する。たとえば、テキスト印刷データについての中間調ビットは、「1」に割り当てられることができ、グラフィックス印刷データについての中間調ビットは「10」に割り当てられることができ、イメージ印刷データについての中間調ビットは「01」に割り当てられることができ、いかなる中間調操作もピクセルに対応する印刷データに適用されないことを示す中間調ビットは、「00」に割り当てられることができる。中間調ビットの割り当ては、中間

調ビットに関連する印刷データの種類について最適化された中間調操作の適用を可能にする。たとえば、中間調ビットは、インチあたり異なるライン数のライン・スクリーン(line screen)を有する中間調アルゴリズムを特定することができる。中間調操作は、印刷技術分野において周知である。中間調に関するさらなる情報は、ISBN 0-262-21009-6(1993年に4版) Ulichney, Rの「Digital Halftoning」に見受けられ、ここで参照により取り入れる。

【0065】併合操作13は、無損失性ラスタ印刷データストリーム、損失性ラスタ印刷データストリーム、中間調印刷データストリーム、併合印刷データストリームを入力として受け取る。これらの印刷データストリームのそれぞれが、8ビットのバイトから形成される。ページ全体のピクセルのそれぞれについてプリントエンジンカラー仕様の1つのカラープレーンを表す印刷データが、無損失性および損失性のラスタ印刷データストリームに含まれる。併合操作13の機能は、最初の画像(前に無損失性および損失性のページ片要素に分離された)が、処理されているカラープレーンについて再構成されるよう、ピクセルごとにこれら2つのラスタ印刷データストリームを組み合わせることである。併合操作13を通る損失性および無損失性のラスタ印刷データの引き続きおこる次のバスが、プリントエンジンカラー仕様の残っているプレーンについて最初の画像を再構成する。ピクセルのそれぞれが、印刷されるページ上の位置に対応する。再構成処理は、無損失性および損失性のラスタ印刷データの空間的同期とみなされ、最初の画像を生成する。さらに、空間的同期を達成するため、無損失性および損失性のラスタ印刷データストリームと中間調の印刷データストリームは、時間的に同期しなければならない。

【0066】印刷データ処理パイプライン15の構成は、印刷データ処理パイプライン15を通る印刷データの通過の前およびその間に実行される。パイプラインのいくつかのパラメータが、各ページ、各カラープレーンのいくつか、各ページ片のいくつかについて構成される。ページあたりのライン数およびラインあたりのピクセル数は、ページごとに構成することができる。さらに、印刷処理パイプライン15により処理されるカラープレーンの数(すなわち、モノクロまたはフルカラーで印刷される画像である)が、ページごとに構成可能である。

【0067】各カラープレーンについてのラスタ印刷データを処理することは、カラー空間変換操作3, 9, 12の構成を必要とする。カラー空間変換操作3, 9, 12の実行に先立ち、出力カラー空間の意図したカラープレーンへの変換に必要なカラー空間変換テーブルがロードされる。さらに、適切な中間調テーブルがカラープレーンを中間調処理するために使用されるよう、中間調操

作14を通る各カラープレーンの処理は中間調操作14の構成を必要とする。たとえば、画像の連続的なカラープレーンを処理するとき、各中間調テーブルは、異なる相対的な観点で選択されたライン・スクリーン(line screen)を使用する中間調アルゴリズムに対応することができる。中間調の技術分野が周知なように、連続的に印刷されたカラープレーンについての異なる相対的な観点で選択されたライン・スクリーンの使用が、印刷加工を生成するカラープレーン間の相互作用を減少させる。

【0068】また印刷データ処理パイプライン15は、ページ片ごとに構成可能なパラメータを有する。画像処理操作1は印刷データ処理パイプライン15を構成することができ、各ページ片について多様な使用可能なパイプライン処理操作を実行する。たとえば、画像処理操作1はページ片ごとについて印刷データ処理操作15を構成し、圧縮操作4、5、伸張操作7、8、10、11、カラー空間変換操作3、9、12または中間調操作14を実行またはバイパスすることができる。さらに、無損失性ページ片要素のピクセルあたりのビット数を、ページごとに1、2、4、8を特定することができる。印刷データ処理パイプライン15の好ましい実施形態では、損失性圧縮操作5で使用される圧縮アルゴリズムが、損失性ページ片要素のピクセルあたり8ビットを必要とする。さらに、画像処理操作1は、ページ片の損失性または無損失性要素がヌルデータ(null data)から形成することができることを指定することができ、対応する併合プレーンの各ビットが同じ値であることを指定することができる。

【0069】図3は、印刷データ処理パイプライン15の好ましい実施形態における簡略化されたハードウェアブロック図である。図3に示される機能ブロック図に接続されるバスは、データバスを表す。図3には、機能ブロック間におけるデータの流れを制御するのに使用される多様な制御ラインは示されていない。好ましい実施形態では、圧縮／伸張操作4、5、7、8、10、11、カラー空間変換操作3、9、12、併合操作13、中間調操作14が、ASIC120で実行される。印刷データ処理パイプライン15の好ましい実施形態は、PCIバスにASIC120を接続するよう構成される。しかし、当該技術分野の当業者には、印刷データ処理パイプライン15が、VESAバスのような他の種類のバスに接続されるよう構成できることがわかるであろう。PCIバスアーキテクチャは、デジタルシステム設計の技術分野において周知であり、ここで詳細には述べない。画像処理操作1は、マイクロプロセッサ132のようなプロセッサにより実行され、PCIバスを介して印刷データ処理パイプライン15と通信を行う。印刷データは、ASIC120への転送およびASIC120からの転送時に、ラスタ印刷データメモリ2および圧縮ラスタ印刷データメモリ6から検索され、およびメモリ2、6にロードされる。

【0070】ASIC120は、PCIバスインターフェース121を介してPCIバスと接続する。PCIバスインターフェース121は、PCIバスインターフェース121の構成に必要なレジスタおよびPCIバスとASIC120間を流れる時に印刷データのバッファリングに必要なレジスタを含む。

【0071】ビデオDMA122のようなダイレクトメモリアクセス・コントローラは、印刷データ処理パイプライン15の多様な機能を実行するASIC120の一部への印刷データの流れおよびASIC120の一部からの印刷データの流れを制御する。ビデオDMAバッファ123は、ビデオDMA122を通過するときに印刷データを一時的に保管し、印刷データがASIC120へおよびASIC120から移動する時、PCIバスとASIC120の多様な機能ブロックの間の印刷データレートを適合させる。ビデオDMAバッファの使用は、PCIバスインターフェース121およびASIC120の残りの部分における機能ブロックの間のデータの流れのレートにおける不整合を補償する。

【0072】いくつかの重要な有利な点が、ASIC120における印刷データ処理パイプライン15の実現から結果として生じる。第1の有利な点は、印刷データ処理パイプライン15についての処理操作が、ASIC120の内部で達成されるという効率的なやり方である。すべての操作をASIC120に置くことにより、印刷データは処理のオーバーヘッドを招くことなく連続した操作間で転送されるが、1つの印刷データパイプライン処理操作の実行に専用の別個の集積回路間のPCIバスに印刷データの転送が生じる場合には処理のオーバーヘッドが生じる。第2の有利な点は、ASIC120の機能ブロックに関して印刷データを選択的にバイパスするようASIC120を構成することができるということである。第3の有利な点は、他の機能ブロックに転送するため、またはメモリに保管するため、機能ブロックの出力を選択的にビデオDMA122にフィードバックすることができるということである。これらの機能は、非常に柔軟性があつて効率的な印刷データ処理パイプラインを作り、印刷データ処理パイプラインは印刷データの種類に依存して印刷データ処理操作を最適化するよう構成される。ASIC120が、可能な印刷データ処理操作の特定のサブセットを実行するよう構成されるので、印刷データに適用される操作が、幅広い種類のやり方で印刷データの特性に適合される。これは、印刷システムが印刷データのすべての可能な種類を最適に処理するようパイプラインを構成することを可能にする。フィードバック機能は、印刷データ処理操作がページ印刷の印刷データを用意するのに一番適した順番で実行されることを可能にする。印刷データ処理パイプライン15の操作が完了すると、その結果がビデオDMA122により次の操作に向けられる。

【0073】PCIバスからロードされた印刷データについて実行する印刷データ処理パイプライン15の操作の

種類に依存して、ビデオDMA122はASIC120の適切な機能ブロックに印刷データを配信するよう構成される。カラー空間変換が必要とされると、印刷データはビデオDMA122によりページ片マネージャ124に送られる。ページ片マネージャ124は、印刷データをビデオDMA122からカラー空間変換器125の適切な損失性または無損失性の入力へ送りこむ。さらにページ片マネージャ124は、無損失性ラスタ印刷データにインターリーブされた併合データおよび中間調データを抽出し、これらのそれぞれを別個のバイト幅チャネルを介して併合ユニット128に送る。印刷データの圧縮または伸張が必要な場合には、印刷データはビデオDMA122により、損失性伸張／圧縮器126または無損失性圧縮／伸張器127の適切な圧縮または伸張の入力に宛てて送られる。併合ユニット128は、カラー空間変換器125から出力される損失性および無損失性のラスタ印刷データストリームについて併合操作13を実行する。ラインバッファ129は、併合操作13で使用される損失性および無損失性のラスタ印刷データと併合および中間調データのバッファリングのためS R A Mを含む。中間調ユニット130は、中間調プレーンで特定されるビットに従ってピクセルのそれぞれに対応して、併合されたラスタ印刷データストリームについて中間調操作14を行う。図3には明示的に示していないが、カラー空間変換器125および中間調ユニット130は、印刷データについて実行するべき操作がカラー空間変換および（または）中間調処理を含まない場合には、これらの機能ブロックに関して印刷データをバイパスするよう構成される。

【0074】図3の印刷データ処理パイプライン15を通る印刷データの流れについて操作が完了した時、印刷データはASIC120からメモリに転送され、後でさらなる処理のためにASIC120に戻る。たとえば、損失性5および無損失性4の圧縮操作がそれぞれ損失性126および無損失性127の圧縮器／伸張器で実行された後、その結果の圧縮ラスタ印刷データは、PCIバスインターフェース121を通ってビデオDMA120に転送され、その後PCIバスを介して圧縮ラスタ印刷データメモリ6に転送される。圧縮ラスタ印刷データが伸張される時は、圧縮ラスタ印刷データが、圧縮ラスタ印刷データメモリ6からPCIバスを介しPCIバスインターフェース121を通ってASIC120に転送され、DMA122により損失性126および無損失性127の圧縮器／伸張器のどちらかまたは両方の伸張入力に送られる。カラー空間変換操作3が損失性および（または）無損失性のラスタ印刷データについて実行された後、その結果のラスタ印刷データはビデオDMA122を通って損失性126圧縮／伸張器および（または）無損失性127の圧縮／伸張器に戻され、またはビデオDMA122およびPCIバスインターフェース121を通ってラスタ印刷データメモリ2に

戻され、または併合操作13を行うため併合ユニット128に送られる。カラー空間変換器125にはセレクタ(selector)が含まれ、損失性または無損失性のカラー空間の変換されたラスタ印刷データを、併合ユニット130の損失性および無損失性の印刷データ入力に接続されるカラー空間変換器125の損失性または無損失性の出力に送る。

【0075】ページ片マネージャ124がラスタ印刷データを管理するので、印刷データ処理パイプライン15のカラー空間変換操作3、9、12、併合操作13、中間調操作14が、一定形式を有するラスタ印刷データのストリームを受け取る。たとえば、1つのみの損失性または無損失性のページ片要素が画像処理操作1で生成されるという状況では、対応する損失性および無損失性のページ片要素は損失性および無損失性のバス上をカラー空間変換器125に送られるよう、ページ片マネージャ124がブランク印刷データの対応する損失性または無損失性のページ片を生成する（適したように）。さらに、1つのみの損失性または無損失性のページ片要素が画像処理操作1により生成される場合では、ページ片マネージャ124が、ページ片要素の各ピクセルに対応する同じビットで形成される対応する併合プレーンを生成する。さらに、同じ中間調処理がページ片の各ピクセルに適用される場合では、ページ片マネージャ124はページ片に対応する中間調プレーンを生成する。ページ片マネージャ124は併合データおよび中間調データを、別個のバイト幅バスを介して併合ユニット128へ送る。

【0076】ページ片マネージャ124は、別個のバイト幅チャネルを介して伸張された損失性および無損失性のラスタ印刷データを受け取る。圧縮損失性ラスタ印刷データが、損失性圧縮／伸張器126により伸張されると、カラー空間の各次元の結果としての8ピクセルx8ピクセルのブロックが、損失性圧縮／伸張器126に含まれる3つの64バイトバッファに保管される。ページ片マネージャ124は、3つの出力バッファから送られる損失性ラスタ印刷データのバイト幅ストリームを受け取る。各ピクセルに対応する伸張損失性ラスタ印刷データのバイトは、カラー空間の各次元に対応する1バイトを有する3つの連続したバイトとして、損失性圧縮／伸張器126の出力バッファから送られる。たとえば、3つの連続したバイトの1番目は「Y」要素に対応し、2番目は「Cb」要素に対応し、3番目は「Cr」要素に対応する。ページ片マネージャ124は、ピクセルについて損失性ラスタ印刷データの3つの対応するバイトを組み立て、それらを損失性ラスタ印刷データの24ビット幅ストリームとしてカラー空間変換器125に送る。

【0077】ページ片マネージャ124は、各無損失性ピクセルについて3つの連続バイトのバイト幅ストリームとして伸張無損失性ラスタ印刷データを受け取る。た

とえば、3つの連続バイトの1番目は「R」要素に対応し、2番目は「G」要素に対応し、3番目は「B」要素に対応する。無損失性圧縮／伸張器127から送られた無損失性ラスタ印刷データにインターリーブ(interleave、はさみこみ)されるのは、併合データおよび中間調データである。インターリーブは、印刷されたページの1ラインに対応する無損失性ラスタ印刷データのバイトが送られ、無損失性ラスタ印刷データのラインに対応する併合データおよび中間調データの必要なバイト数がこれに続くよう行われる。ページ片マネージャ124は併合データおよび中間調データを抽出し、これらを別個のチャネルを介して併合ユニット128に送る。ページ片マネージャ124はピクセルについての無損失性ラスタ印刷データの3つの対応するバイトを組み立て、これらを無損失性ラスタ印刷データの24ビット幅ストリームとしてカラー空間変換器125に送る。

【0078】いくつかのケースでは、ビデオDMA122が圧縮のステップをバイパスし、ラスタ印刷データを直接ページ片マネージャ124に送る場合がある。これは、圧縮および圧縮ラスタ印刷データメモリ6への保管をすることなく、ラスタ印刷データのページがプリントエンジン16に送られる時になされる。この場合、ラスタ印刷データは、いかなる多様なフィードバックパスを使うことなく、直線的に印刷データ処理パイプライン15を通って流れる。また、ラスタ印刷データが直接ビデオDMA122からページ片マネージャ124に送られることもでき、カラー空間変換および中間調処理のような引き続き起る次の印刷データ処理パイプライン15の処理操作を通り、適切な圧縮器に送るためビデオDMA122を通るフィードバックパス上に戻り、その後メモリに保管するためビデオDMA122に圧縮ラスタ印刷データ戻す。

【0079】無損失性ラスタ印刷データは、各ピクセルのカラー空間の次元あたり1ビット、2ビット、4ビットまたは8ビットにより特定される。無損失性ラスタ印刷データをカラー空間変換器125に送ることに先立ち、ページ片マネージャ124は1ビット、2ビットおよび4ビットの表示を、各ピクセル表示の次元あたりフルの8ビットに変換する。ページ片要素のそれぞれに適用される印刷データ処理パイプライン操作の異なる種類を有するのと同様に、ラスタ印刷データのパラメータはページ片要素の間で変更することができる。ページ片マネージャ124は、一定形式のラスタ印刷データのシームレスなストリームが（ここでシームレスとは、ページ片要素間のラスタ印刷データの形式の違いが取り除かれ、印刷データ処理パイプライン15を通るラスタ印刷データの流れのレートにおいてほとんど変動が無いことを意味する）、印刷データ処理パイプライン操作のダウンストリーム（下流）に配信されるのに必要な操作を実行する。

【0080】実行されそうな印刷データ処理操作の多くの種類の特定の例をあげることは、ASIC120で実現される印刷データ処理パイプライン15のかなりの多様性をより良く例示する。CMYKエンジンカラー仕様で表現されるラスタ印刷データを使用してページが印刷される第1のケースを考えてみる。ここでは、いかなる圧縮／伸張操作も行われず、中間調処理がラスタ印刷データに適用される。損失性および無損失性のラスタ印刷データが、一回に1カラープレーン分、PCIバスインターフェース121を通ってシステムメモリから転送される。ビデオDMA122は損失性および無損失性のラスタ印刷データを、ページ片マネージャ124を介して、印刷データをバイパスするよう構成されているカラー空間変換器125に送る。併合ユニット128は損失性および無損失性のラスタ印刷データについて併合操作13を行い、併合された印刷データを中間調操作14のために中間調ユニット13に送る。中間調処理されたラスタ印刷データは、プリントエンジンインターフェース131を通ってプリントエンジン16に送られる。この第1のケースでは、印刷データ処理パイプライン15は併合操作13および中間調操作14のみを行うよう構成された。

【0081】第2のケースでは、印刷データ処理パイプライン15は損失性および無損失性の圧縮ラスタ印刷データについて伸張操作10、11を実行し、カラー空間変換操作12、併合操作13、および中間調操作14を実行するよう構成される。損失性および無損失性の圧縮ラスタ印刷データは、システムメモリからPCIバスインターフェース121を通って転送される。ビデオDMA122は損失性および無損失性の圧縮ラスタ印刷データを、伸張のために損失性126および無損失性127の圧縮／伸張器に送る。伸張無損失性および損失性のラスタ印刷データは、プリントエンジン16の4バスのそれぞれについて1つのカラープレーンに変換するため、カラー空間変換器125に送られる。損失性および無損失性のカラー空間のカラー空間変換器125から変換されたラスタ印刷データ出力は、併合操作13を行うため併合ユニット128に送られる。中間調操作14が併合ユニット128からの併合された印刷データ出力について実行される。中間調操作の結果は、PCIバスインターフェース121を通ってシステムメモリに送るため、ビデオDMA122に送り戻される。この一連の動きは連続して4回発生し、プリントエンジン16の各カラープレーンについて印刷データを生成する。この第2のケースでは、印刷データ処理パイプライン15は、後で印刷をするためにシステムメモリに処理された印刷データを戻す。ビデオDMA122がすべての印刷データを2回取り扱う必要があったが（ロードの間1回と、保管の間1回）、複数の操作がPCIバスの使用またはマイクロプロセッサ132の干渉なくASIC120の印刷データについて実行された。

【0082】3番目のケースでは、印刷データ処理パイプライン15は、カラー空間変換操作、中間調操作、その後に無損失性圧縮操作を実行するよう構成される。無損失性ラスタ印刷データは、システムメモリからPCIバスインターフェース121を通って転送される。ビデオDMA122は無損失性ラスタ印刷データをページ片マネージャ124を通ってカラー空間変換器125へ送り、プリントエンジン16のカラー空間のプレーンへの変換を受ける。ページ片マネージャ124は対応するブランクの損失性ラスタ印刷データおよび併合印刷データを生成する。ブランク損失性ラスタ印刷データは、カラー空間変換器125に関してバイパスされる。併合ユニット130は、ページ片マネージャ124により生成された併合印刷データを使用して、無損失性および損失性のラスタ印刷データについて併合操作を実行する。中間調ユニット130は、併合されたラスタ印刷データについて中間調操作14を実行する。中間調ユニット130からのラスタ印刷データ出力は、ビデオDMA122に送られる。その後無損失性圧縮のため、ビデオDMA122はラスタ印刷データを無損失性圧縮／伸張器127に送る。圧縮ラスタ印刷データはビデオDMAに戻され、PCIバスインターフェース121を通って保管のためシステムメモリに送られる。この一連の動きは連続して4回発生し、プリントエンジン16の各カラーブレーンについて印刷データを生成する。第2のケースとは対照的に、実行された印刷データ処理操作の順番が変更された。これは、印刷データ処理パイプライン15が、印刷データの処理を最適化するよう構成されることを示す。

【0083】印刷データ処理パイプラインの従来技術が、一般用のマイクロプロセッサを使用して実現され、多様なカラー空間変換、圧縮／伸張、併合、中間調操作が行われてきた。これらの操作を行うファームウェアを実行するマイクロプロセッサのスピードは、専用ハードウェアを使用して達成されるものより遅い。さらに、これらの操作を成し遂げるために一般用のマイクロプロセッサを使用するのは、システムバスを介してマイクロプロセッサおよびメモリ間を転送する中間印刷データのために著しいデータ転送のオーバーヘッドを生じさせる。システムバスを介する転送が非常に大きいことが、一般用のマイクロプロセッサを用いて実現された印刷データ処理パイプラインの処理能力を著しく低下させる。

【0084】印刷データ処理パイプラインは、専用の集積回路を使用して実現され、印刷データ処理パイプラインの特定の機能を達成してきた。カラー空間変換および圧縮／伸張のような操作は、個々の集積回路で実現されてきた。しかし、一般用のマイクロプロセッサはなお、印刷データ処理パイプラインの操作が実行されるとき、多様な集積回路およびメモリ間を印刷データを移動させる必要がある。これは、専用ハードウェアの使用による

処理時間におけるいくつかの改良を提供するが、データ転送のオーバーヘッドはなお、パフォーマンスの制約要因である。ASIC120の印刷データ処理パイプライン15の実現は、ASIC120における複数パイプライン操作を実行することにより、このデータ転送のオーバーヘッドを回避する。

【0085】併合ユニット130の好ましい実施形態は、損失性および無損失性のラスタ印刷データストリームから選択し、併合ビットを使用して1つのデータストリームを形成し、最初の画像を再構成する。しかし、併合ユニット130で使用される技術は、同様のやり方で併合されたデータの各ユニットについての複数の併合ビットを使用して2つより多いデータストリームを組み合わせるのにも有効であることを理解すべきである。さらに、併合されたデータストリームは、損失性および無損失性のラスタ印刷データストリームとは別の方法で、ラスタ印刷データまたはラスタ印刷データの区画とは別のデータの種類を含むことができる。たとえば、ラスタ印刷データストリームは、ラスタ印刷データのピクセルへの中間調操作の適用に依存して複数のストリームに分割することができる。中間調操作を受けるラスタ印刷データのこれらのピクセルは、中間調処理されたラスタ印刷データについて最適化される圧縮操作を使用して圧縮される。中間調処理を受けなかったラスタ印刷データのピクセルは、異なる圧縮操作を受ける。ラスタ印刷データストリームの他の可能な分割は、ピクセルの各カラー空間を表すのに使用されるビット数に基づく。たとえば、黒および白の情報を含むページ領域は、カラー画像を含むページ領域よりピクセルあたりのビットを少なくして表される。最も極端な場合では、ページのある領域はピクセル表示あたりフルカラー24ビットをもち、ページの他の領域は、ピクセル表示あたりバイナリの黒および白の1ビットをもつことができる。ラスタ印刷データのピクセルあたりフルカラー24ビットのストリームおよびラスタ印刷データのピクセルあたり1ビットのストリームを別個に処理することは、ストリームのそれぞれについて圧縮および中間調操作を最適化することを可能にする。ラスタ印刷データストリームをさらに分割できることは、ページを表示するのに使用される解像度に基づく。たとえば、細かい特徴を含まない画像の領域は、600dpiのような低い解像度で印刷される。非常に細かい特徴を含む画像の領域は、1200dpiのような高い解像度で印刷される。このケースは、鮮明な境界を有する一定のグレイスケール領域(gray scale region)が存在するときに発生することができる。高い解像度のラスタ印刷データストリームおよび低い解像度のラスタ印刷データストリームを別個に処理することは、ラスタ印刷データストリームのそれぞれについて中間調および圧縮操作を最適化することを可能にする。

【0086】印刷データ処理パイプライン15の好まし

い実施形態は、併合ユニット130を使用して併合操作13をデジタルデータについて実行するが、感光ドラム(photoconductor drum)の表面の損失性および無損失性のラスタ印刷データの現像(連続バス)を通ってプリントエンジン16の感光ドラムの表面へ併合操作を実行することも可能なことに注意すべきである。この併合操作は、感光ドラムが現像される時、損失性および無損失性のラスタ印刷データの間の効率的なOR操作である。さらに、併合プレーンを使用する代わりに、損失性および無損失性のラスタ印刷データの間にOR操作が実行され、これらストリームから選択されるべきラスタ印刷データを特定するよう、デジタルデータについて併合操作13を実行することもできる。

【0087】図4は、併合ユニット128により実行される操作を図で表す。セクション20から23のそれぞれが、SRAMラインバッファ129の一部であり、各セクションが2つのバッファに対応し、併合ユニット128に入る印刷データストリームの4種のうちの1つを保持する。バッファリングが必要なのは、十分な印刷データを必要な時に印刷データストリームのそれぞれから使用でき、併合ユニット128のラスタ印刷データの処理量を下げることなく併合操作13を実行するためである。併合ユニット128は各併合ビットを処理するので、ラスタ印刷データのバイトは、処理されているピクセルに割り当てるため、無損失性または損失性の印刷データストリームの両方から使用できなければならない。さらに、中間調ビットが、処理されているピクセルとの空間的関係を維持するために使用できなければならぬ。無損失性印刷データストリームについて、ピクセルが印刷されるページに連続的に置かれた時にプリントエンジンの2つの走査線全体をカバーするよう、無損失性ラインバッファ21に十分なバイト数が保管される。無損失性ラインバッファ21を形成する2つのバッファに無損失性ラスタ印刷データの2ラインを保管することにより、無損失性ラスタ印刷データは「ピンポン」のやり方で無損失性ラインバッファ21を通って移動することができ、高い処理量を維持する。

【0088】印刷データ処理パイプライン15の好ましい実施形態における損失性ラスタ印刷データについてのデータ圧縮アルゴリズムの操作は、圧縮走査に使用されるセルを形成するため、損失性ページ片要素のピクセルを分割することにより動作する。印刷データ処理パイプラインの好ましい実施形態で選択されたセルのサイズは、幅が8ピクセル、高さが8ピクセルである。伸張が圧縮損失性ラスタ印刷データに実行されると、幅8ピクセルで高さ8ピクセルのブロックに対応するのにラスタ印刷データの十分なバイトが、損失性伸張操作11の出力として生成される。カラー空間変換器125からの損失性ラスタ印刷データ出力の十分なバッファリングを確実にするため、圧縮損失性ラスタ印刷データの伸張がブ

ロックごとになされるので、損失性ラインバッファ20は2ラインのブロックを保管する容量をもつ。2ラインのブロックは、ピクセルの十分な数に相当し、16のプリントエンジンの走査線全体をカバーする。損失性ラインバッファ20を形成する2つのバッファに損失性ラスタ印刷データの2ラインのブロックを保管することにより、損失性ラスタ印刷データは「ピンポン」のやり方で損失性ラインバッファを通って動くことができ、高い処理能力を維持する。併合ラインバッファ22および中間調ラインバッファ23はそれぞれ、十分な併合ビットと中間調ビットを含み、ピクセルの2走査線を処理する。従って、併合ラインバッファ22の記憶容量は、無損失性ラインバッファ21の1/8であり、中間調のラインバッファ23の記憶容量は、無損失性ラインバッファ21の1/4である。

【0089】印刷データマルチプレクサ24の入力は、損失性ラインバッファ20および無損失性ラインバッファ21の出力に接続される。併合ラインバッファ22の出力は、印刷データマルチプレクサ24の出力のため、入力ラスタ印刷データの選択を制御する。処理を受けているピクセルの併合ビットの状況に基づいて、印刷データマルチプレクサ24は、損失性ラスタ印刷データストリームまたは無損失性ラスタ印刷データストリームのどちらかから処理されているピクセルについてラスタ印刷データを選択する。印刷データマルチプレクサ24の出力からの併合されたラスタ印刷データストリームは、2つの入力ラスタ印刷データストリームの組み合わせなので、最初の画像のピクセル間における空間的同期が、再構成された画像に維持される。無損失性および損失性のラスタ印刷データストリームの両方の対応するピクセルに前に割り当てられた中間調ビットが、併合ユニット128に中間調ビットが入った時に存在したラスタ印刷データへのピクセル毎の空間的同期を維持する。

【0090】図5は、併合ユニット128の高レベルブロック図である。それぞれ8ビットのバイトのストリームから成る損失性ラスタ印刷データストリーム30および無損失性ラスタ印刷データストリーム31が、それぞれ損失性入力バッファ32および無損失性入力バッファ33に保管される。それぞれが8ビットのバイトから成るストリームに形成される併合ビット34のストリームおよび中間調ビット35のストリームは、併合入力バッファ36および中間調入力バッファ37にそれぞれ保管される。入力バッファ32、33、36、37は、印刷データストリームが入力バッファ32、33、36、37から送られる時に、これらの印刷データストリームのそれぞれが8ビット幅から32ビット幅に変換されるよう設計される。

【0091】カラー空間変換器インターフェース・コントローラ38は、入力バッファ制御機能を含み、入力バッファ32、33、36、37を通る4つの印刷データ

ストリームの転送を管理する。この制御機能は、カラー空間変換器125へハンドシェーキング信号(handshaking signals)を送り、および変換器125からハンドシェーキング信号を受け取る。これらのハンドシェーキング信号は、カラー空間変換器125から入力バッファ32、33、36、37への印刷データの流れを制御する。これらのハンドシェーキング信号から、カラー空間変換器インターフェース・コントローラ38は、許可信号(enable signals)を生成し、入力バッファ32、33、36、37が印刷データの適切な種類(損失性、無損失性、中間調、併合)を入力バッファ32、33、36、37のうちの対応する1つへロードすることを許可する。カラー空間変換器125から受取られたハンドシェーキング信号は、損失性および無損失性のラスタ印刷データが、損失性32および無損失性33の入力バッファに転送するのに使用可能である時を示す信号を含む。またカラー空間変換器インターフェース・コントローラ38は、併合および中間調印刷データが、併合36および中間調37の入力バッファに転送可能である時を示すハンドシェーキング信号を受け取る。また、カラー空間変換器インターフェース・コントローラ38からカラー空間変換器125およびページ片マネージャ124に送られるハンドシェーキング信号は、入力バッファ32、33、36、37のそれぞれが入力印刷データを受け取る用意がある時を示す信号を含む。

【0092】入力バッファ32、33、36、37のそれからの32ビット幅データストリーム40、41、42、43出力は、入力バッファマルチプレクサ39に送られる。カラー空間変換器インターフェース・コントローラ38は、入力バッファマルチプレクサ39を通じて入力バッファ32、33、36、37からの印刷データストリームの流れを制御する。カラー空間変換器インターフェース・コントローラ38により生成された制御信号47は、SRAMインターフェース・コントローラ44に転送するため32ビット幅データストリーム40、41、42、43のうちの1つを選択する。カラー空間変換器インターフェース・コントローラ38に含まれる状況機構(state machine)(別個には示されていない)は、制御信号47を生成し、入力バッファ32、33、36、37からの印刷データストリームのどれが、SRAMインターフェース・コントローラ44に転送するため入力バッファマルチプレクサ39により選択されるかを判断する。

【0093】SRAMインターフェース・コントローラ44およびカラー空間変換器インターフェース・コントローラ38はそれぞれ、必要なハンドシェーク信号45を生成し、入力バッファマルチプレクサ39を介して印刷データをSRAMインターフェース・コントローラ44に転送する。SRAMインターフェース・コントローラ44はハンドシェーク信号を生成し、カラー空間変換

器インターフェースコントローラ38に、SRAMインターフェース・コントローラ44が印刷データを受け取る準備があるということを示す。カラー空間変換器インターフェース・コントローラ38は、入力バッファマルチプレクサ39から送られた印刷データの種類(損失性、無損失性、中間調、併合)を識別する制御信号を提供する。SRAMインターフェース・コントローラ44により受取られた印刷データは、SRAMラインバッファ129に保管される。SRAMラインバッファ129は、印刷データの種類(損失性、無損失性、中間調、併合)のそれぞれに対応するセクションに分割される。SRAMラインバッファ129のラインバッファ20から23は、図4に示されるこれらに対応する。SRAMラインバッファ129の印刷データの保管は、十分な印刷データを連続した併合操作を実行するのに使用することができるということを保証する。SRAMインターフェース・コントローラ44により受取られた印刷データは、カラー空間変換器インターフェース・コントローラ38からの制御信号45により判断されたように、SRAMラインバッファ129の適切なセクションに保管される。SRAMラインバッファ129に保管された印刷データは、SRAMインターフェース・コントローラ44にロードされるので、印刷データを併合操作13を実行するのに使用することができる。印刷データは、1つの双方向32ビット幅のバス49を介してSRAMラインバッファに保管され、またそこからロードされる。このバス49は図5に2つの单方向バスとして示されているが、併合ユニット128の好ましい実施形態は、1つの双方向バスを使用する。SRAMインターフェース・コントローラ44はアドレスを生成し、アドレスバス50を介してSRAMラインバッファ129の記憶場所にアクセスする。

【0094】SRAMラインバッファ129からSRAMインターフェース・コントローラ44にロードされる印刷データは、併合操作を実行するのに備えて、出力バッファ51、52、53、54に転送される。出力バッファ51、52、53、54は、損失性出力バッファ51、無損失性出力バッファ52、併合出力バッファ53、中間調出力バッファ54を含む。中間調出力バッファ54および併合出力バッファ53は、それぞれシフトレジスタから形成される。シフトレジスタを使用することにより、中間調印刷データおよび併合印刷データの32ビットワードを、対のビットに(中間調印刷データ)または個々のビットに(併合印刷データ)分割することが可能となり、これらは通常関連するピクセルに対応するラスタ印刷データと突き合わせられる。SRAMインターフェース・コントローラから出力バッファ51、52、53、54への印刷データの転送は、中間調インターフェース・コントローラ55およびSRAMインターフェース・コントローラ44の間で制御信号56により

管理される。制御信号56は、出力バッファ51、52、53、54のそれぞれが印刷データの対応する種類（損失性、無損失性、中間調、併合）を受け取る準備がある時を示す信号を含み、SRAMインターフェース・コントローラ44が出力バッファ51、52、53、54に転送するのに使用できる印刷データを有する時を示す信号を含む。

【0095】中間調インターフェース・コントローラ55は制御信号57を生成し、出力バッファ51、52、53、54を通る印刷データの転送を制御する。出力バッファ51、52、53、54のそれぞれへの入力印刷データストリームは32ビット幅である。損失性ラスタ印刷データ出力バッファ51および無損失性ラスタ印刷データ出力バッファ52からの損失性および無損失性のラスタ印刷データストリーム出力は、それぞれ8ビット幅である。制御信号57は許可信号を含み、適切な印刷データの種類が、対応する出力バッファ51、52、53、54へロードされることを許可する。損失性51および無損失性52のラスタ印刷データ出力バッファのそれぞれに含まれるのはマルチプレクサであり（図5には示されていない）、印刷データマルチプレクサ24への転送のため、32ビット幅の損失性および無損失性のラスタ印刷データストリームに含まれる4つの8ビットのバイトのうちの1つを選択する。制御信号57は、損失性出力バッファ51および無損失性出力バッファ52の一部であるマルチプレクサを制御するのに必要な信号を含む。中間調インターフェース・コントローラ55により生成された制御信号57は、32ビット幅の中間調および併合データストリームを、対応する中間調データ出力バッファ54および併合データ出力バッファ53にロードするのに必要な信号を含む。前述したように、中間調データ出力バッファ54および併合データ出力バッファ53のそれぞれが、シフトレジスタを含む。さらに、中間調データ出力バッファ54および併合データ出力バッファ53のそれぞれがマルチプレクサ（図5には示されていない）を含み、制御信号57に含まれる信号により判断されたように、右シフト出力または左シフト出力のどちらかを選択する。またこの信号は、シフトレジスタが併合データおよび中間調データを動かす方向を制御する。両面モード(duplex mode)印刷のため、併合データおよび中間調データを動かすシフトレジスタの方向を制御することが必要である。両面モード印刷では、ラスタ印刷データがプリントエンジン16に配信される順番が、ページの一番上と一番下を基準として、ページの表側と裏側の印刷の間で逆方向になる。

【0096】印刷データマルチプレクサ24は、バイト幅の損失性および無損失性のラスタ印刷データストリームを入力として受け取る。併合出力バッファ53からの1ビット幅併合データストリーム出力が、併合ユニット128からの出力のため入力ラスタ印刷データストリーム

ムのうちの1つの選択を制御する。併合ユニット128の多様なバッファを通じて移動された時、併合データストリームの併合ビットの相対的な順番、およびそれぞれのラスタ印刷データストリームの損失性および無損失性のラスタ印刷データの相対的な順番を維持することにより、印刷データマルチプレクサ24により実行される併合操作13が、ピクセルごとに最初の（オリジナルの）ページを正確に再構成する。デジタル設計の技術分野の当業者には、この明細書を理解した後に、印刷データマルチプレクサ24の機能を実行するのに必要な論理回路を設計する知識を所有するであろう。

【0097】SRAMラインバッファ129は、損失性および無損失性のラスタ印刷データ、併合データ、中間調データの間の時間的および空間的な同期の確立を可能にする。伸張損失性ラスタ印刷データが、8ピクセル幅で高さ8ピクセルのブロックで伸張するために、伸張損失性ラスタ印刷データは併合ユニット128に配信され、他の印刷データへの同期からはずれてSRAMラインバッファ129に保管される。印刷データがSRAMラインバッファ129から移動する（離れる）方法が、時間的および空間的同期を確立する。SRAMラインバッファ129の使用を必要としない併合ユニットの実現が可能であることに注意すべきである。損失性および無損失性のラスタ印刷データストリームおよび併合データが、時間的および空間的に同期した方法で併合ユニットに与えられると、併合ユニット128の内部における印刷データストリームのバッファリングの必要性が除去される。時間的および空間的に同期した印刷データストリームを受け取るのが必要な併合ユニットの実現のため、必要ならば印刷データストリームのバッファリングは併合ユニットの外部で実行される。さらに、損失性および無損失性のラスタ印刷データストリームおよび併合データストリームの間の時間的および空間的同期により、印刷データマルチプレクサを使用して機能する併合ユニットは著しく簡略化される。

【0098】併合データ出力バッファ53は、併合データのバイト幅ストリームを、損失性または無損失性のラスタ印刷データに同期した1ビット幅ストリーム（対応するピクセルを表す）に変換する。併合ユニット128の好ましい実施形態は、損失性および無損失性のラスタ印刷データのバイトから選択するため1ビットを使用するが、複数ビット幅ストリームを、2つより多いラスタ印刷データストリームから選択するのに使用することもでき、ラスタ印刷データストリームは8ビットより多くを使用して、カラー空間の各次元を表すこともできる。

【0099】図6は、損失性入力バッファ32の入力側配置の簡略化された概念図であり、入力バッファ32、33、36、37のそれぞれの操作を表す。典型的な損失性入力バッファ32は、8個の8ビットのデータラッシュ60～67から形成される。8ビット入力損失性ラス

タ印刷データストリーム68が、8個のデータラッチ60～67の8個の「DATA」入力へ送られる。8個のデータラッチ60～67の「EN」入力に接続された8個の3入力ANDゲート69～76は、制御信号（カラー空間変換器インターフェース・コントローラ38により生成される）を解読し、入力印刷データストリームのバイトのロードのため8個のデータラッチ60～67の1つを選択する。解読ロジックは、カラー空間変換器インターフェース・コントローラ38の一部である。図6に示されるように、8個のデータラッチ60～67は、それぞれが4個のデータラッチを有する2つのバンクに区切られる。印刷データストリームの4つの連続的に転送されたバイトが2つのバンクの片方にロードされ、印刷データストリームの次の4つの連続的に転送されたバイトが2つのバンクの他方にロードされるように、制御信号の解読がなされる。印刷データストリームの4バイトの連続的に転送されるグループが、データラッチの2つのバンクのうちの1つに交互にロードされるように、入力バッファ32、33、36、37へのロードは、入力バッファ32、33、36、37を通る印刷データの流れを維持する。2つのバンクのうちの片方が入力印刷データでロードされる間、2つのバンクのうちの他方からの印刷データの転送が発生する。

【0100】図7は、損失性入力バッファ32の出力側配置の簡略化された概念図であり、入力バッファ32、33、36、37のそれぞれの操作を表す。8個のデータラッチ60～67の「OE」入力が固定論理レベルに接続され、損失性入力バッファ32に保管される損失性ラスタ印刷データが、各クロックサイクルの立ち上がりエッジでデータラッチ60～67から出力される。損失性入力バッファ32の2つのバンクのそれぞれの4つの8ビット出力が、2つの32ビット幅の印刷データを形成する。これら2つの32ビット幅印刷データストリームは、マルチプレクサ70の入力に接続される。これら2つのうちの1つ、32ビット幅印刷データストリームが、入力バッファマルチプレクサ39への入力のために選択される。選択は、カラー空間変換器インターフェース・コントローラ38により生成された制御信号を使用して、マルチプレクサ70により実行される。この制御信号は、入力バッファ32、33、36、37の2つのバンクのうちどれがフル(full)で、入力バッファマルチプレクサ39に送られるのに含まれる印刷データを有する用意があるかを検出する。2バンクのそれぞれに保管される4バイトのグループのうちの1つが交互に選択されるので、入力バッファ32、33、36、37からの印刷データの転送は、入力バッファ32、33、36、37を通る印刷データの流れを維持し、入力バッファマルチプレクサ39への印刷データの流れを維持する。印刷データの4バイトが2つのバンクのうちの片方から転送される間、2つのバンクの他方への印刷データの4バ

イトのロードが発生する。

【0101】図8は、損失性出力バッファ51および無損失性出力バッファ52の入力側配置の簡略化された概念図を示す。損失性51および無損失性52出力バッファのそれぞれが、4個の8ビットデータラッチ80～95の2つのバンクを含む。併合ユニット128の内部の32ビット幅の損失性ラスタ印刷データストリームおよび無損失性ラスタ印刷データストリームが、それぞれ4つの8ビット幅損失性または無損失性のラスタ印刷データストリームに分割され、各データラッチ80～95のそれぞれの「DATA」入力に送られる。データラッチ80～95のそれぞれの「EN」入力は、損失性および無損失性のラスタ印刷データのデータラッチ80～95へのロードを制御する。4つのANDゲート96～99は、中間調インターフェース・コントローラ55の一部であり、中間調インターフェース・コントローラ55により生成される制御信号を解読し、データラッチ80～95のロードを制御する。損失性および無損失性のラスタ印刷データが、データラッチ80～95の1対のバンクにロードされる間、損失性および無損失性のラスタ印刷データは、データラッチ80～95の他の1対のバンクから送られる。このバッファリング方法の使用は、損失性および無損失性出力バッファ51、52を通る印刷データの流れを維持する。

【0102】図9は、損失性51および無損失性52出力バッファについてのデータラッチ80～83、88～91の損失性および無損失性の1対のバンクの出力側配置の簡略化した概念図である。損失性51および無損失性52出力バッファのデータラッチ84～87、92～95の他の2つのバンクの出力側も同様に実現される。損失性51および無損失性52出力バッファは、32ビット幅入力損失性および無損失性のラスタ印刷データストリームを、8ビット幅の損失性および無損失性のラスタ印刷データストリームに変換する。8個のデータラッチ80～83、88～91の「OE」入力が、固定論理レベルに接続され、これらのデータラッチ80～83、88～91に保管される損失性および無損失性のラスタ印刷データが、各クロックサイクルの立ち上がりエッジで出力される。マルチプレクサ100、101は、印刷データマルチプレクサ24への入力のため、それぞれのデータラッチ80から83、88から91からの8ビット出力のうちの1つを選択するのに使用される。それぞれのデータラッチ80～83、88～91からの8ビット出力が、それぞれのマルチプレクサ100、101により選択されるので、それぞれの損失性および無損失性のラスタ印刷データストリームの8ビットのバイトの相対的な順番が維持される。これは、印刷データマルチプレクサ24により実行される併合操作13が正確に最初のページを再構成するのに必要である。マルチプレクサ100、101を制御するのに使用される信号は、中間

調インターフェース・コントローラ55により生成される。併合出力バッファ53による併合ビット出力のストリームは、印刷データ・マルチプレクサ24により、損失性および無損失性のラスタ印刷データストリームからの8ビットのバイトの選択を制御する。印刷データ・マルチプレクサ24から出されるラスタ印刷データは8ビット幅であり、併合された損失性および無損失性のラスタ印刷データストリームから成る。

【0103】図10は、中間調出力シフトレジスタ110および併合出力シフトレジスタ111の簡略化された概念図である。32ビット幅中間調印刷データストリーム112および32ビット幅併合印刷データストリーム113が、それぞれの出力シフトレジスタ110、111にロードされる。中間調インターフェース・コントローラ55から出力シフトレジスタ110、111のそれぞれの「LD」入力への信号は、中間調および併合印刷データのロードを制御する。シフトレジスタ110、111の使用により、中間調印刷データおよび併合印刷データが調整されたバイトを、それぞれ2ビット幅および1ビット幅の印刷データのストリームに分割することができる、これらの2つのデータストリームの相対的な順番が、損失性および無損失性のラスタ印刷データストリームに関して維持される。

【0104】ANDゲート114、115は、中間調インターフェース・コントローラ55に含まれ、それぞれのシフトレジスタ110、111からの併合印刷データおよび中間調印刷データのシフト・アウトを制御する。シフトレジスタ110、111の「EN」入力がアサートされると、それぞれのレジスタが、併合印刷データおよび中間調印刷データをシフト・アウトする。シフトレジスタ110、111への「DIR」入力のレベルに依存して、併合および中間調印刷データは、クロックの立ち上がりエッジで「SLD」または「SRD」出力からシフト・アウトされる。併合および中間調印刷データの相対的順番が損失性および無損失性のラスタ印刷データストリームを基準として維持されるよう、「DIR」入力はレジスタ110、111の併合および中間調印刷データのシフトの方向を制御する。併合ユニット128の内部でそれぞれの32ビット幅印刷データストリームに形成される時に、32ビットの最上位および最下位のビットに対して、併合および中間調ビットがどのように並べられるかに依存して、損失性および無損失性の印刷データストリームへの空間的同期を維持するよう「DIR」入力が制御される。

【0105】シフトレジスタ110からシフト・アウトされた中間調印刷データストリームは2ビット幅であり、シフトレジスタ111からシフト・アウトされた併合データストリームは1ビット幅である。マルチプレクサ116、117は、シフトレジスタ110、111の「DIR」入力に接続される同じ信号で制御され、マル

チプレクサ116、117から出力するためシフトレジスタ110、111の「SRD」または「SLD」出力を選択する。マルチプレクサ117からの併合印刷データ出力の1ビット幅ストリームは、印刷データ・マルチプレクサ24の制御入力に接続され、ピクセルごとに損失性または無損失性のラスタ印刷データストリームから選択する。併合ユニット128からの出力印刷データは、バッファ・マルチプレクサ24からの8ビット幅の併合されたラスタ印刷データストリーム出力を含み、マルチプレクサ116からの2ビット幅の中間調印刷データストリーム出力を含む。

【0106】本発明のいくつかの実施形態を例示し説明したが、当該技術分野の当業者には、本発明から離ることなく様々な改良がなされることができることが明らかであろう。

【0107】本発明は例として次の実施態様を含む。

(1) 複数のデータ要素からそれぞれ形成される複数のデータストリーム(201、202)を別個に処理するためのデータ処理パイプライン(200、15)であって、前記複数のデータストリーム(201、202)により定められ、前記データ要素に対応する併合データ要素から形成される併合データストリーム(209)を使用するデータ処理パイプラインは、前記複数のデータストリーム(201、202)の第1のデータストリーム(201)を受け取る第1の入力を有し、該第1のデータストリームの前記データ要素について第1の変換を実行し、第1の変換されたデータストリーム(205)を生成する第1のパイプライン処理ユニット(203)と、前記複数のデータストリーム(201、202)の第2のデータストリーム(202)を受け取る第2の入力を有し、該第2のデータストリームの前記データ要素について第2の変換を実行し、第2の変換されたデータストリーム(206)を生成する第2のパイプライン処理ユニット(204)と、前記第1の変換されたデータストリーム(205)、前記第2の変換されたデータストリーム(206)および前記併合データストリーム(209)を受け取るよう構成され、前記第1の変換されたデータストリームと前記第2の変換されたデータストリームを、前記併合データストリームを使用して出力データストリーム(208)に併合する併合ユニット(207、128)と、を備えるデータ処理パイプライン。

【0108】(2) 上記(1)におけるデータ処理パイプライン(200、15)において、前記複数のデータストリーム(200、201)の前記第1のデータストリーム(201)の前記データ要素は、選択的に無損失性データおよび圧縮無損失性データを含み、前記複数のデータストリームの前記第2のデータストリーム(202)は、選択的に損失性データおよび圧縮損失性データを含み、前記併合データ要素は併合データを含み、前記第1のパイプライン処理ユニット(203)は無損失性伸張出力および無損失性圧縮出力を有する無損失性圧縮／伸張器(127)を含み、該無損失

性圧縮／伸張器は前記併合データを受け取り、圧縮併合データを生成する機能および前記圧縮併合データを受け取り伸張併合データを生成する機能を含み、前記第1の変換されたデータストリーム(205)は、選択的に圧縮無損失性データおよび伸張無損失性データを含み、前記第2のパイプライン処理ユニットは、損失性伸張出力および損失性圧縮出力を有する損失性圧縮／伸張器(126)を含み、前記第2の変換されたデータストリームは選択的に圧縮損失性データおよび伸張損失性データを含み、前記出力データストリーム(208)は併合された無損失性および損失性データを含み、前記併合ユニット(207, 128)は、前記無損失性データ、前記損失性データ、前記併合データを受け取り、前記併合された無損失性および損失性データを生成し、前記伸張無損失性データ、前記伸張損失性データおよび前記伸張併合データを受け取り、前記併合された無損失性および損失性データを生成する機能を含む上記(1)のデータ処理パイプライン。

【0109】(3) 上記(2)におけるデータ処理パイプライン(200, 15)において、前記無損失性圧縮出力および前記損失性圧縮出力に接続され、前記無損失性、前記圧縮無損失性、前記伸張無損失性データ、前記損失性、前記圧縮損失性、前記伸張損失性データ、および前記併合、前記圧縮併合、前記伸張併合データのうちの少なくとも1つを受け取る第3の入力を有し、前記無損失性、前記圧縮無損失性、前記伸張無損失性データ、および前記併合、前記圧縮併合、前記伸張併合データのうちの少なくとも1つを前記無損失性圧縮／伸張器(127)に送り、前記損失性、前記圧縮損失性、前記伸張損失性データうちの少なくとも1つを前記損失性圧縮／伸張器(126)に送るよう構成されたダイレクトメモリアクセス・コントローラ(122)と、前記無損失性伸張出力、前記損失性伸張出力、および前記ダイレクトメモリアクセス・コントローラに接続され、第1の無損失性出力、第1の損失性出力および前記併合ユニット(128)に接続された併合データ出力を有するページ片マネージャ(124)と、前記第1の無損失性出力および前記第1の損失性出力に接続され、併合ユニットに接続された第2の無損失性出力および第2の損失性出力を有するカラー空間変換器(125)であって、前記第1および第2の変換のそれぞれがカラー空間変換を含むカラー空間変換器と、を備える上記(2)のデータ処理パイプライン。

【0110】(4) 上記(3)におけるデータ処理パイプラインにおいて、前記カラー空間変換器(125)は、前記カラー空間変換器を通る前記無損失性データ、前記伸張無損失性データ、前記損失性データ、前記伸張損失性データをバイパスする機能を含み、前記カラー空間変換器(125)は、前記無損失性データおよび前記損失性データから選択するため、および前記伸張無損失性データおよび前記伸張損失性データから選択するため、前記ダイレクトメモリアクセス・コントローラ(122)に接続され

たセレクタ出力を有するセレクタを含み、前記無損失性および損失性データは、印刷システムにおける使用のため無損失性ラスタ印刷データおよび損失性ラスタ印刷データをそれぞれ含み、前記併合された無損失性および損失性のデータは、併合された無損失性および損失性ラスタ印刷データを含み、前記ページ片マネージャ(124)は、次の処理操作のため、前記無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データ、前記損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データの少なくとも1つをフォーマットする機能を含み、前記印刷システムは、電子写真方式(electrophotographic)印刷システム(16)を含む上記(3)のデータ処理パイプライン。

【0111】(5) 複数の印刷データ要素からそれぞれ形成される複数の印刷データストリーム(201, 202)を別個に処理するための印刷データ処理パイプライン(200, 15)であって、前記複数の印刷データストリーム(201, 202)により定められ、前記印刷データ要素に対応する併合データ要素から形成される併合データストリーム(209)を使用する印刷データ処理パイプラインは、前記複数の印刷データストリーム(201, 202)の第1の印刷データストリーム(201)を受け取る第1の入力を有し、該第1の印刷データストリームの前記印刷データ要素について第1の変換を実行し、第1の変換された印刷データストリーム(205)を生成する第1のパイプライン処理ユニット(203)と、前記複数の印刷データストリーム(201, 202)の第2の印刷データストリーム(202)を受け取る第2の入力を有し、該第2の印刷データストリームの前記印刷データ要素について第2の変換を実行し、第2の変換された印刷データストリーム(206)を生成する第2のパイプライン処理ユニット(204)と、前記第1の変換された印刷データストリーム(205)、前記第2の変換された印刷データストリーム(206)および前記併合データストリーム(209)を受け取るよう構成され、前記第1の変換された印刷データストリームと前記第2の変換された印刷データストリームを、前記併合データストリームを使用して出力印刷データストリーム(208)を併合する併合ユニット(207, 128)と、を備える印刷データ処理パイプライン。

【0112】(6) 上記(5)における印刷データ処理パイプライン(200, 15)において、前記併合データ要素は選択的に併合データおよび伸張併合データを含み、前記複数の印刷データストリームの前記第1のストリームの前記データ要素は、選択的に無損失性ラスタ印刷データおよび圧縮無損失性ラスタ印刷データを含み、前記複数の印刷データストリームの前記第2のストリームの前記データ要素は、選択的に損失性ラスタ印刷データおよび圧縮損失性ラスタ印刷データを含み、前記出力印刷データストリーム(208)は、併合された無損失性および損失性のラスタ印刷データを含み、前記第1のパイプライン

処理ユニット(203)は、前記無損失性ラスタ印刷データから前記圧縮無損失性ラスタ印刷データを生成し、前記圧縮無損失性ラスタ印刷データから伸張無損失性ラスタ印刷データを生成する無損失性圧縮／伸張器(127)を含み、前記第2のパイプライン処理ユニット(204)は、前記損失性ラスタ印刷データから前記圧縮損失性ラスタ印刷データを生成し、前記圧縮損失性ラスタ印刷データから伸張損失性ラスタ印刷データを生成する損失性圧縮／伸張器(126)を含み、前記第1の変換されたデータストリーム(205)は、選択的に前記無損失性ラスタ印刷データ、前記圧縮無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データを含み、前記第2の変換されたデータストリーム(206)は、選択的に前記損失性ラスタ印刷データ、前記圧縮損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データを含み、前記無損失性圧縮／伸張器(127)は、無損失性伸張出力および無損失性圧縮出力を含み、前記併合データを受け取り、圧縮併合データを生成する機能および前記圧縮併合データを受け取り前記伸張併合データを生成する機能を含み、前記損失性圧縮／伸張器(126)は、損失性伸張出力および損失性圧縮出力を含む上記(5)の印刷データ処理パイプライン。

【0113】(7) 上記(6)における印刷データ処理パイプラインはさらに、前記無損失性圧縮出力および前記損失性圧縮出力に接続され、前記無損失性、前記圧縮無損失性、前記伸張無損失性ラスタ印刷データ、前記損失性、前記圧縮損失性、前記伸張損失性ラスタ印刷データ、および前記併合、前記圧縮併合、前記伸張併合データのうちの少なくとも1つを受け取る入力を有し、前記無損失性、前記圧縮無損失性、前記伸張無損失性データ、および前記併合、前記圧縮併合、前記伸張併合データのうちの少なくとも1つを前記無損失性圧縮／伸張器(127)に送るよう構成され、前記損失性、前記圧縮損失性、前記伸張損失性ラスタ印刷データのうちの少なくとも1つを前記損失性圧縮／伸張器(126)に送るよう構成されたダイレクトメモリアクセス・コントローラ(122)と、印刷データから前記無損失性ラスタ印刷データ、前記損失性ラスタ印刷データおよび前記併合データを生成し、ダイレクトメモリアクセス・コントローラの前記入力に送るプロセッサ(132)と、前記無損失性伸張出力、前記損失性伸張出力および前記ダイレクトメモリアクセス・コントローラに出力され、第1の無損失性出力、第1の損失性出力、前記併合ユニットに接続された併合データ出力を有し、前記無損失性ラスタ印刷データ、前記損失性ラスタ印刷データおよび前記併合データ、前記伸張無損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データおよび前記伸張併合データのうちの少なくとも1つを次の処理操作のためフォーマットする機能を含むページ片マネージャと、前記第1の無損失性出力および前記第1の損失性出力に接続され、前記併合ユニットに接続された第2の無損失性出力および第2の損失性出力

を有し、前記カラー空間変換器を通る前記無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データ、前記損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データをバイパスする機能を含み、前記無損失性ラスタ印刷データおよび前記損失性ラスタ印刷データから選択するため、または前記伸張無損失性ラスタ印刷データおよび前記伸張損失性ラスタ印刷データから選択するため、前記ダイレクトメモリアクセス・コントローラに接続されるセレクタ出力を有するセレクタを含むカラー空間変換器と、を備える上記(6)の印刷データ処理パイプライン。

【0114】(8) プリンタであって、複数の印刷データ要素からそれぞれ形成される複数の印刷データストリーム(201, 202)を生成し、前記複数の印刷データストリームにより定められ、前記印刷データ要素に対応する併合データ要素から形成される併合データストリームを生成するプロセッサ(132)と、前記プロセッサから前記複数の印刷データストリームを受け取るよう構成されたダイレクトメモリアクセス・コントローラ(122)と、前記ダイレクトメモリアクセス・コントローラ、前記第1のパイプライン処理ユニット(203, 127)から前記複数の印刷データストリームのうちに第1の印刷データストリームを受け取る第1の入力を有し、該第1の印刷データストリームの前記印刷データ要素について第1の変換を実行し、第1の変換された印刷データストリームを生成する第1のパイプライン処理ユニットと、前記ダイレクトメモリアクセス・コントローラ、前記第2のパイプライン処理ユニット(204, 126)から前記複数の印刷データストリームのうちに第1の印刷データストリームを受け取る第2入力を有し、該第2の印刷データストリームの前記印刷データ要素について第2の変換を実行し、第2の変換された印刷データストリームを生成する第2パイプライン処理ユニットと、前記第1の変換された印刷ストリーム、前記第2の変換された印刷ストリームおよび前記併合データストリームを受け取るよう構成され、前記第1の変換された印刷データストリームおよび前記第2の変換された印刷データストリームを、前記併合データストリームを使用して出力印刷データストリームに併合する併合ユニット(207, 128)と、を備えるプリンタ。

【0115】(9) 上記(8)におけるプリンタにおいて、前記併合データ要素は、選択的に併合データおよび伸張併合データを含み、前記複数の印刷データストリームの前記第1の印刷ストリームの前記データ要素は、選択的に無損失性ラスタ印刷データおよび圧縮無損失性ラスタ印刷データを含み、前記複数の印刷データストリームの前記第2の印刷ストリームの前記データ要素は、選択的に損失性ラスタ印刷データおよび圧縮損失性ラスタ印刷データを含み、前記出力印刷データストリームは、併合された無損失性および損失性ラスタ印刷データを含み、前記第1のパイプライン処理ユニットは、前記圧縮

無損失性ラスタ印刷データを前記無損失性ラスタ印刷データから生成し、伸張無損失性ラスタ印刷データを前記圧縮無損失性ラスタ印刷データから生成する無損失性圧縮／伸張器を含み、前記第2のパイプライン処理ユニットは、前記圧縮無損失性ラスタ印刷データを前記無損失性ラスタ印刷データから生成し、伸張無損失性ラスタ印刷データを前記圧縮無損失性ラスタ印刷データから生成する損失性圧縮／伸張器を含み、前記第1の変換されたデータストリームは、選択的に前記無損失性ラスタ印刷データ、前記圧縮無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データを含み、前記第2の変換されたデータストリームは、選択的に前記損失性ラスタ印刷データ、前記圧縮無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データを含み、前記無損失性圧縮／伸張器(127)は、無損失性伸張出力および無損失性圧縮出力を含み、前記併合データを受け取って圧縮併合データを生成し、前記圧縮併合データを受け取って前記伸張併合データを生成する機能を含み、前記損失性圧縮／伸張器(126)は損失性伸張出力および損失性圧縮出力を含む上記(8)のプリンタ。

【0116】(10) 上記(9)におけるプリンタにおいて、前記無損失性伸張出力、前記損失性伸張出力および前記ダイレクトメモリーアクセス・コントローラに接続され、第1の無損失性出力、第1の損失性出力および前記併合ユニットに接続された併合データ出力を有し、前記無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データ、前記損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データ、前記併合データおよび前記伸張併合データの少なくとも1つを次の処理操作のためにフォーマットする機能を含むページ片マネージャと、前記第1の無損失性出力および前記第1の損失性出力に接続され、前記併合ユニットに接続された第2の無損失性出力および第2の損失性出力を有し、前記カラー空間変換器を通じ前記無損失性ラスタ印刷データ、前記伸張無損失性ラスタ印刷データ、前記損失性ラスタ印刷データ、前記伸張損失性ラスタ印刷データをバイパスする機能を含み、前記無損失性ラスタ印刷データおよび前記損失性ラスタ印刷データから選択するため、および前記伸張無

損失性ラスタ印刷データおよび前記伸張損失性ラスタ印刷データから選択するため、前記ダイレクトメモリーアクセス・コントローラに接続されたセレクタ出力を有するセレクタを含むカラー空間変換器と、を備える上記(9)のプリンタ。

【0117】

【発明の効果】本発明によるデータ処理パイプラインによれば、データ要素に基づいて、圧縮比と印刷品質の最適な組合せを適用することができるようになる。また、

10 印刷データ処理パイプラインにおける操作を適切な位置におくことにより、印刷品質をさらに良くし、必要なメモリ量を減少させることができるようになる。

【図面の簡単な説明】

【図1】印刷データ処理パイプラインの高レベル図。

【図2】カラープリンタにおける印刷データ処理パイプラインの好ましい実施形態を通じ印刷データの流れの概念図。

【図3】印刷データ処理パイプラインの好ましい実施形態のハードウェア・ブロック図。

20 【図4】併合操作の概念図。

【図5】併合ユニットの高レベルブロック図。

【図6】損失性入力バッファの入力側の概念図。

【図7】損失性入力バッファの出力側の概念図。

【図8】損失性および無損失性出力バッファの入力側の概念図。

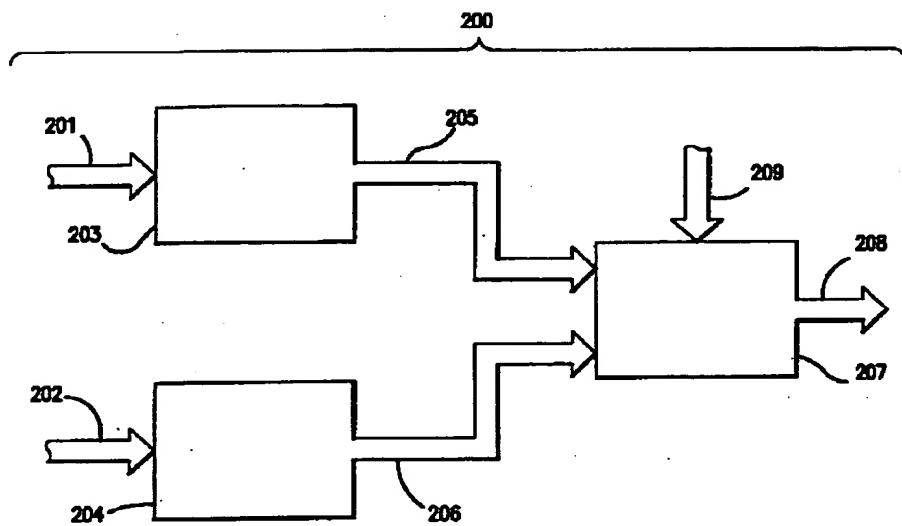
【図9】損失性および無損失性出力バッファの出力側の概念図。

【図10】中間調および併合印刷データストリームに使用されるシフトレジスタの概念図。

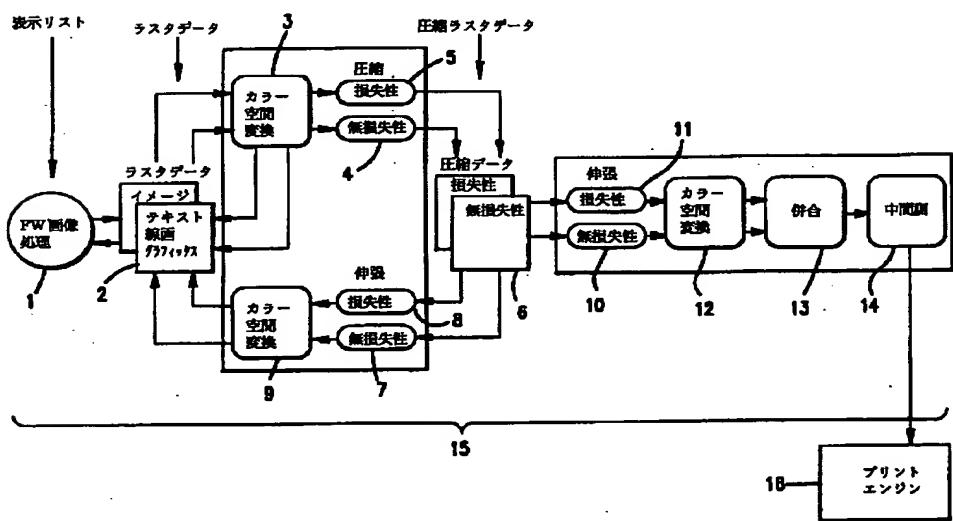
30 【符号の説明】

15、200	印刷データ処理パイプライン
201	第1のデータストリーム
202	第2のデータストリーム
205	第1の変換されたデータストリーム
206	第2の変換されたデータストリーム
128、207	併合ユニット
208	出力データストリーム
209	併合データストリーム

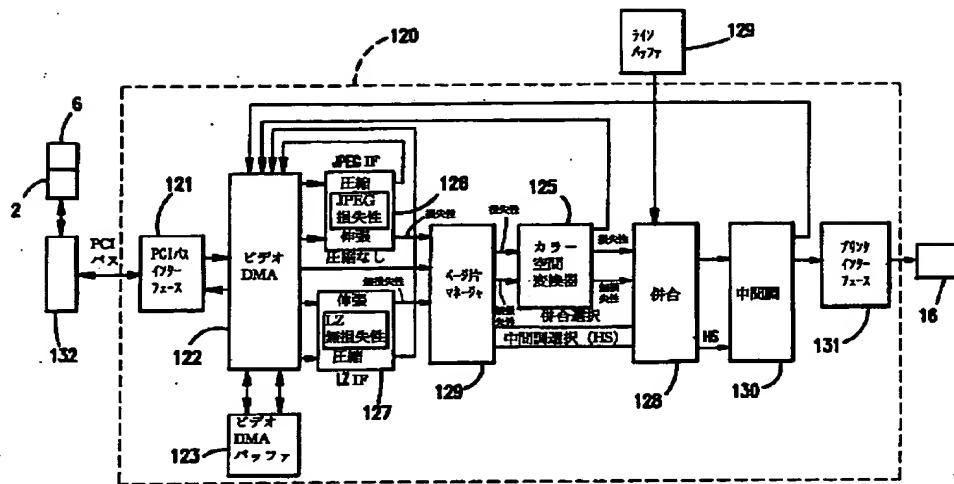
【図1】



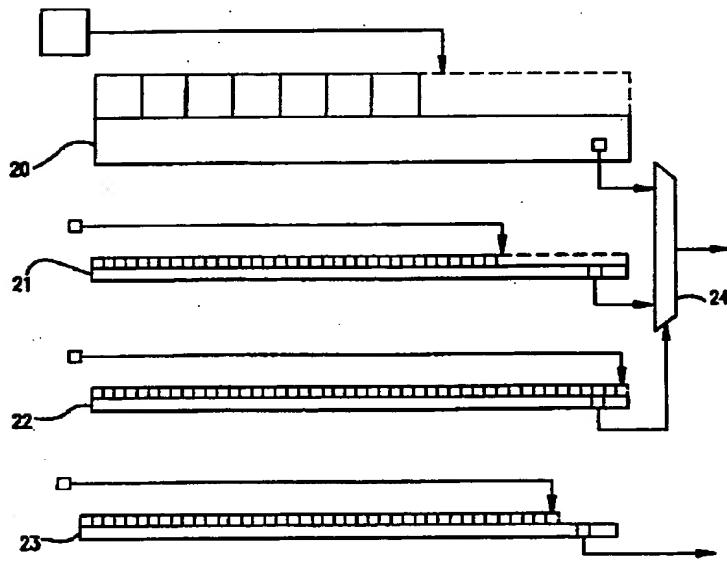
【図2】



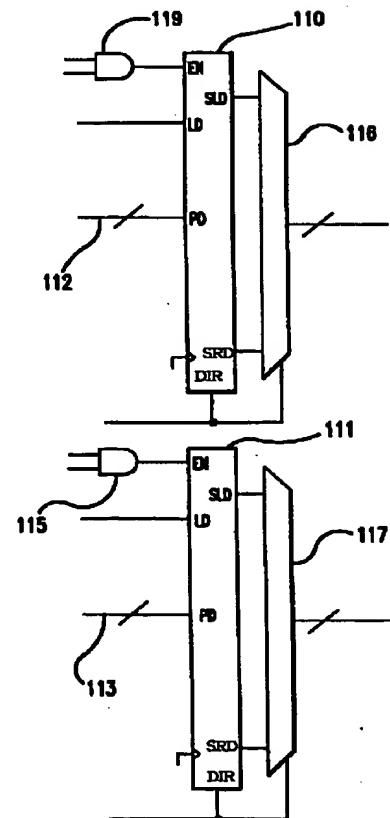
【図3】



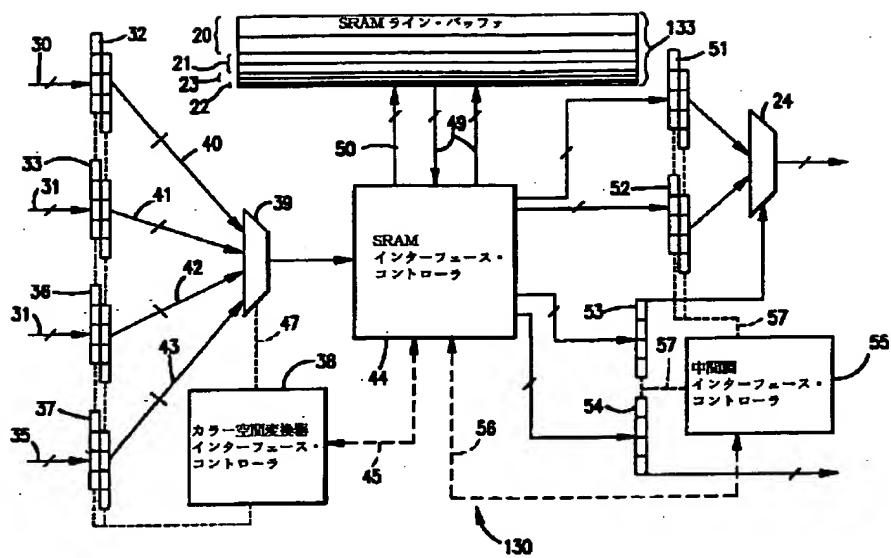
【図4】



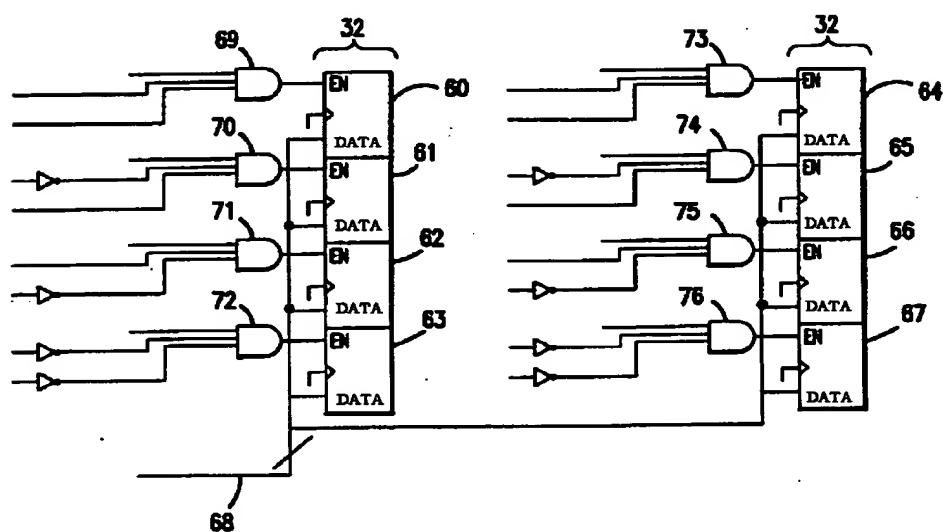
【図10】



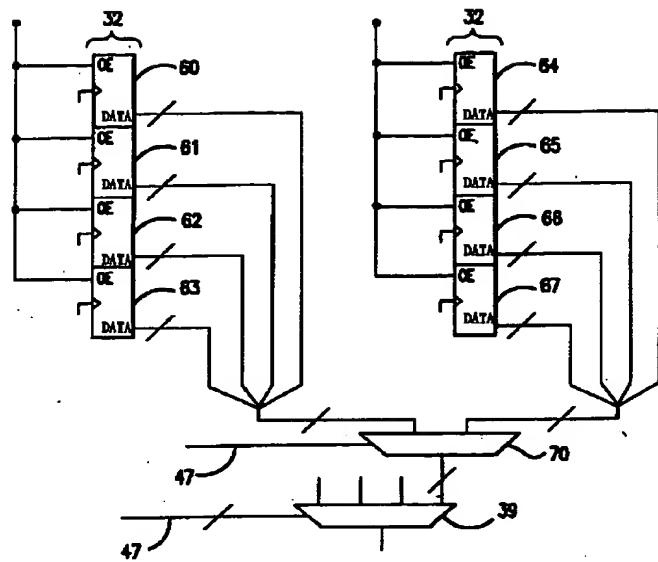
【図5】



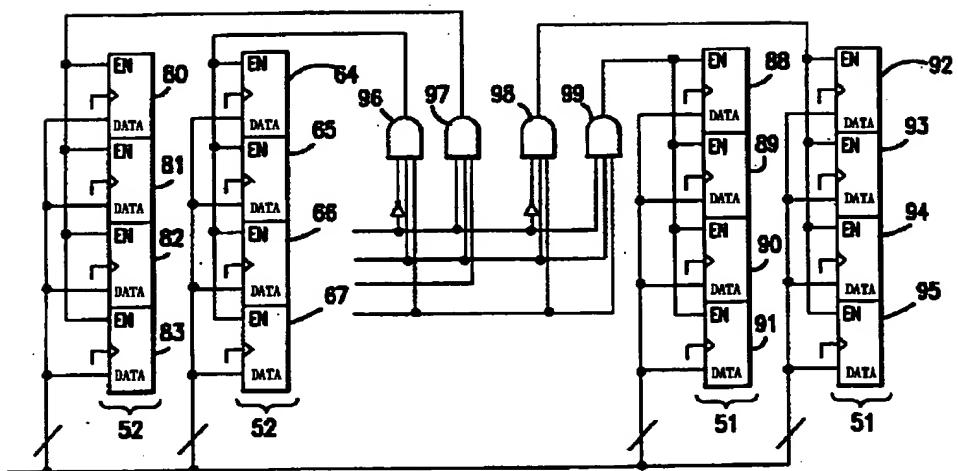
【図6】



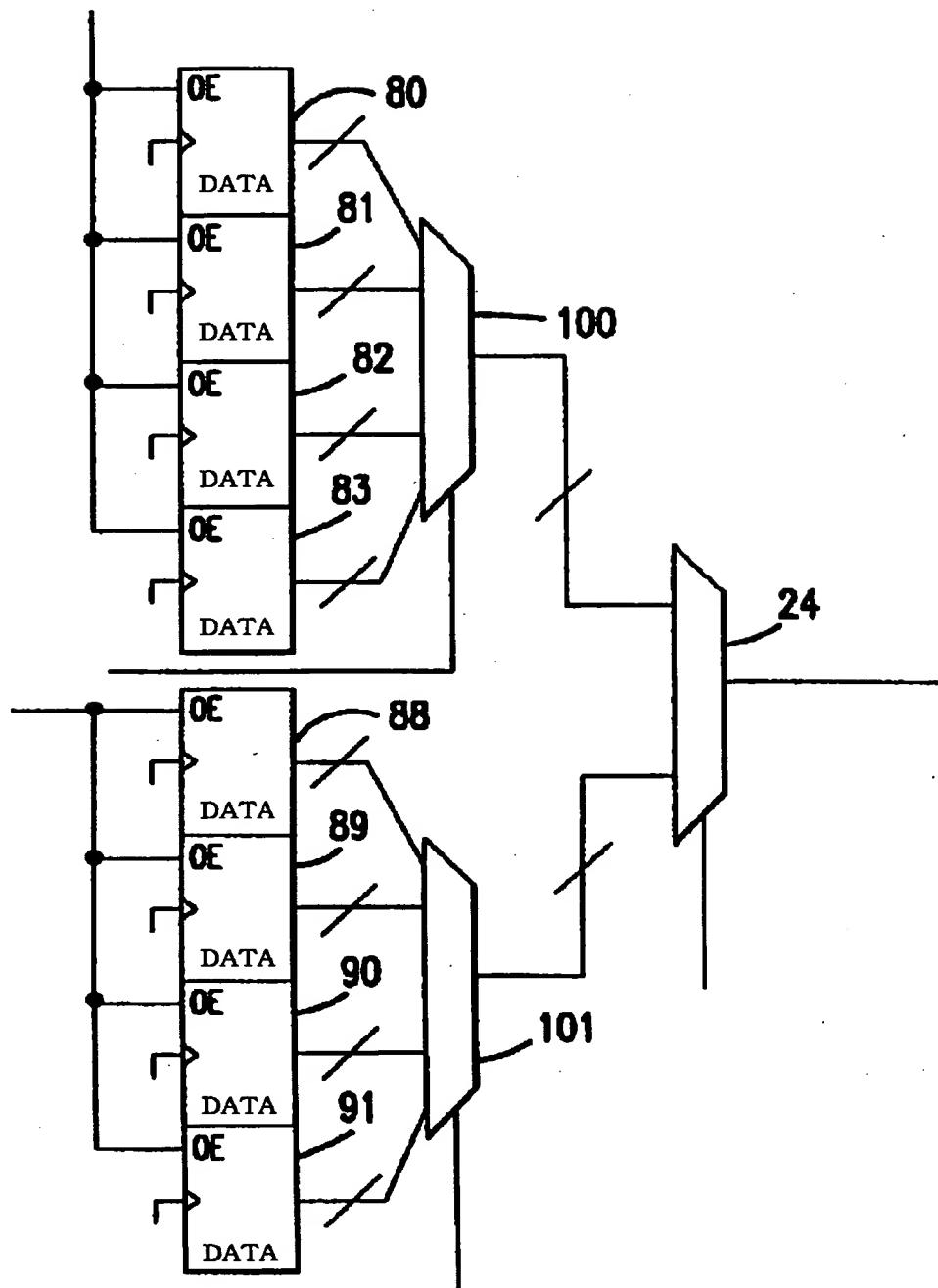
【図 7】



【図 8】



【図9】



フロントページの続き

(51) Int.C1.⁶ 識別記号
H 0 4 N 1/411

F I
H 0 4 N 1/40 Z

(72) 発明者 ジェームス・アール・ノッティンガム
アメリカ合衆国83713アイダホ州ボイジー、
ノース・ジャガー・アベニュー 4741

(72) 発明者 ダグラス・ヘインズ
アメリカ合衆国83318アイダホ州バーレー、
リバー・ラン・エステーツ 23

JPO and NCIPI are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. **** shows the word which can not be translated.
3. In the drawings, any words are not translated.

CLAIMS

[Claim(s)]

[Claim 1] He is a data-processing pipeline for processing separately two or more data streams formed from two or more data elements, respectively. It is set by said two or more data streams, and the merge data stream formed with the merge data element corresponding to said data element is used. It has the 1st input which receives the 1st data stream of two or more of said data streams. The 1st pipeline processing unit which generates the data stream from which 1st conversion was performed about said data element of this 1st data stream, and the 1st was changed, It has the 2nd input which receives the 2nd data stream of two or more of said data streams. The 2nd pipeline processing unit which generates the data stream from which 2nd conversion was performed about said data element of this 2nd data stream, and the 2nd was changed, It is constituted so that said data stream from which the 1st was changed, said data stream from which the 2nd was changed, and said merge data stream may be received. The data-processing pipeline having the merge unit by which the data stream from which said data stream from which the 1st was changed, and said 2nd [the] were changed is merged to an output data stream using said merge data stream.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention more specifically relates to a print-data processing pipeline's configuration in a printer about processing of data.

[0002]

[Description of the Prior Art] The print-data pipeline of a printer performs many actuation about the print data inputted into a pipeline in preparation for printing. These actuation includes compression of print data, elongation of print data, conversion of color space, and halftone processing (halftoning). The specific sequence that the class of actuation performed and its actuation are performed changes depending on the memory which can use the class of print data included in a pipeline, the function of a print engine, and a printer. A text, a line drawing (line art), an image, and graphics can be included in the class of print data which can go into a pipeline. Processing actuation with the various implementation of a pipeline in the conventional technique is performed by the processor under control of firmware. Depending on the class of print data included in a pipeline, many possible firmware routines are performed if needed, and perform the above-mentioned actuation.

[0003] It depends on the class of print data included in a pipeline for the detail of actuation of the print-data compression performed. For example, by the kind of print data like image print data of a certain kind, the print-data condensing routine which brings about a certain informational loss (loss) is permitted. The debasement of the printed output is not accepted by the class of these print data. the "condensing routine in which informational loss is not accepted with the printed output -- it is no losing nature (visually lossless) visually" -- it is called a system. However, it is important for the quality of the output printed about the class of other print data like a text or a line drawing that the print-data condensing routine to be used does not bring about informational loss.

[0004] A data compression / elongation system is known in the technical field concerned, encodes the stream of a digital data signal to a compression digital code signal, and decrypts and returns a compression digital code signal to the first data. A data compression says a certain processing which is going to change the data of a given format into the format alternative to tooth spaces fewer than the beginning being sufficient. The purpose of a data compression system is saving storage capacity required holding the given contents of digital information. When this digital information is the digital display of an image or a text, a data compression system is classified into two common molds. That is, there is nothing with that (lossy: loss nature) with a loss (lossless: no losing nature). (In addition, lossy and lossless call it respectively irreversible and reversible.) The system of no losing nature has what is called reversible (reciprocity). In order for a data compression system to have a reversible property (property), there is also no deterioration or loss of any information, it must be able to decode and the compressed data must be able to be returned to re-elongation or the form which is the beginning. Moreover, decryption data and the first data must be what it is the same and cannot be distinguished at all mutually. Therefore, the reversible property is synonymous with the thing which is used by the information theory and which does not have a noise (noise) strictly.

[0005] Some applications do not need to be strictly based on a reversible property. As stated above, one of such the specific applications is the case where an image data is treated. Since it is not necessarily easy to sense human being's eye in a noise, the deterioration or loss informational [some of] between compression and elongation is permitted. The system which does this informational loss is made to call it a "loss nature data compression system."

[0006] The important criteria in the design of a data compression system are compressive efficiency, and this is characterized by the compression ratio. A compression ratio (compression ratio) is a ratio of the data size of the incompressible format broken by size of compressed format. In order for data to be compressible, data must contain redundancy. Compressive efficiency is determined by how a compressive procedure uses the redundancy of input data effectively. By usual computer by which data are kept, redundancy happens by both generating (it is (like a general word and the record field of a blank)) from which the use which is not the homogeneity of each notation display (for example, a figure, a cutting tool, an alphabetic character), and a series of notations arise frequently.

[0007] It is given by the printer and a data compression system offers sufficient performance about the data rate (data rate) accepted. The rate into which data are compressed is determined by the processing rate of the input data of a compression system. Sufficient performance is inevitable, in order to maintain the attained data rate and for the processed data to prevent interruption of printing by not being usable. Therefore, a data compression elongation system must have sufficient data band width of face in order

not to have a bad influence on the whole system.

[0008] usually, the speed of the random access memory which is used for the performance of a data compression elongation system keeping compression, count required for elongation, and a statistical data, and draws compression processing, and a system element which is similar to this -- marginal attachment ****. This is applied especially when a compression elongation system is realized by firmware, and it leads so that firmware may perform data compression elongation processing for the central-process unit for general. In such a system, the performance of a compression equipment is characterized per input character under compression by the required number of processor cycles. Performance becomes better as the number of cycles is small. the solution approach of firmware -- the compression elongation speed of firmware -- marginal attachment ****. This is because firmware elongates each cutting tool using some central-process unit cycles. Therefore, in order to gather elongation speed, firmware processing was usually adjusted so that a compression ratio might be dropped.

[0009] The data compression procedure for general is known in the conventional technique. three suitable procedures and Huffman -- law (Huffman), the wardrobe toll method (Tunstall), and Lempel - a jib -- there is a method (Lempel-Ziv). one of the data compression procedures first developed for general [first] -- Huffman -- it is law. if it states briefly -- Huffman -- law carries out the map of the segment of the overall length of a notation to variable-length WORD. the wardrobe toll method -- the variable-length segment of a notation -- fixed-length binary WORD -- a map -- carrying out -- Huffman -- it applies to law. The wardrobe toll procedure as well as the Huffman procedure needs to know the probability of source data beforehand. Moreover, until can satisfy this need of saying that it knows beforehand, to some extent by using the suitable version which accumulates statistical strength processing (statistic strength processing) of data.

[0010] Lempel - a jib -- a procedure carries out the map of the variable-length segment of a notation to variable-length binary WORD. It is asymptotically the optimal when there is no constraint in an input or an output segment. An input data train is analyzed in this procedure by the segment materialized appropriately. Each of a segment consists of exact copies of the first part of the input train added with one new notation from input data. The copy performed is long as much as possible, and it is not restrained by that it is in agreement with the already analyzed segment. The symbolic language showing an output segment includes the information which consists of the die length of a sign, and the pointer to a new notation the place where the first copy part starts. Lempel - The further technique about the data compression technique of a jib is looked at by U.S. Pat. No. 4558302, and takes in by reference here.

[0011] Although the above-mentioned data compression procedure is no effective losing nature procedure for general, some specific classes of redundancy are compressed using other approaches. One of such the losing nature approaches of no generally learned as run length coding (run length encoding, RLE) is well suitable in a graphic form image data. In RLE, each characters of a series of are encoded as the count field which added the identifier of the repeated character. Usually, since it is the need, this coding is not used for the character of two pieces putting the mark on (run) in the ream of each character by the ream of characters fewer than 2 or it. However, in treating the graphic form image expressed with a digital data format, a ream with the same big character exists in given Rhine of arbitration, and RLE performs a compression procedure efficient about such information.

[0012] All the above-mentioned data compression procedures are greatly dependent on the redundancy of data, and it attains a remarkable compression ratio. Since the input data lacks in a certain regular redundancy about the kind of data of a certain kind in these procedures as for clearly disadvantageous one, it is that the output compressed rather than the input actually becomes large. the technical field of printing -- such "compression -- imperfect (incompressible)" data are generated simply. The kind of an image of a certain kind is classified as either "a systematic dither (ordered dither)" or "error diffusion (error diffused)." The image (it is also called "a cluster (cluster)") of a systematic dither is a halftone (half-tone) image, and includes the gray (gray) display of halftone over a whole page. As for such an image, it is common to express the redundancy of partial data, and as stated above, it has turned to no losing nature technique of data coding.

[0013] However, the image (it is also called "distribution (dispersed)") of error diffusion hardly expresses redundancy with those data, but the different compression approach is required for it by them. The print data showing a photograph (photographic) image give another example of print data with low redundancy. As a result, dealing with image data of such [already] a printer was not completed, and it is carrying out use of one data compression device of a page printer. The title "Page Printer Having Adaptive Data Compression For Memory Minimization" of U.S. Pat. No. 5479587 is shown by Cambell and others, and is taken in by reference here. According to this, the page printer is following through various compression technology, and is going to fit the limited memory size fewer than size required for all the pages of the printed data. In the application mentioned above, when there is little memory and it cannot print an image, 1st "Mode M" compression technology is used. The attempt in which a block will be compressed is made by using this technique, using RLE for each line, and encoding the change delta produced from the line within a block in a line. Although "Mode M" compression technology can print a page, when it does not succeed in offering sufficient compression ratio, the 2nd attempt is made using LZW mold compression. When not succeeding in finally obtaining sufficient big compression ratio to which the compression technology of the LZW base can print a page, a loss nature compression procedure is used.

[0014]

[Problem(s) to be Solved by the Invention] In processing of raster print data, before generating the page printed, various actuation is performed about raster print data. A data compression, color space conversion, and actuation like halftone are included in the actuation which can be performed before generating the page printed. In processing of raster print data, there is a case where the various sections of a page are processed the optimal using the data compression of a different class, color space conversion, and halftone actuation, frequently. In order to enable it to perform actuation of various raster print-data processings the optimal about the suitable section of a page, the problem which faced the optimal processing of raster print data and which arises well was division of the raster print data which form a page.

[0015] For example, the amount of memory required to keep the raster print data corresponding to a page is considered. A printer enlarges the consistency (dot per inch) of a dot location. The function of a gray scale (gray scale) is added (in order to decide the level of a gray scale, many numbers of bits are used per pixel). the function of color printing -- containing (an additional bit being needed per pixel from monochrome printing) -- memory required to keep the data used for printing a page reaches by 32

times the memory needed for the monochrome printer of the same resolution. In order that a color printer may enable it to use more appropriate memory size, it is usually used for a data compression technique lowering the initial complement of memory. However, the class from which raster print data differ is compressed the optimal using different compression technology, respectively. For example, about the raster print data corresponding to the section of a page including an image, a compression ratio and the optimal combination of printing quality are attained by using loss nature compression technology. However, about the raster print data corresponding to the section of the page containing a text, a compression ratio and the optimal combination of printing quality are attained by using no losing nature compression technology. Therefore, the data-processing pipeline who enables application of the optimal mold of data-processing actuation is required for processing of each data element.

[0016]

[Means for Solving the Problem] The data-processing pipeline for processing separately each of two or more data streams formed from the cutting tool of print data or two or more data elements like WORD enables optimization of data pipeline processing actuation about each data element. A data-processing pipeline is set by two or more data streams, and uses the merge data stream formed from the merge data element corresponding to a data element. For example, two or more data streams can be formed by dividing a source-data stream like a printing data stream based on each property of a data element for every data element. Depending on the property, a source-data stream can be divided the the best for many data streams from two. In the case of the data-processing pipeline used for processing raster print data, the stream of raster print data can be divided on pixel level. For example, the pixel relevant to the loss nature field of a page can form a loss nature raster printing data stream, and the pixel relevant to no losing nature field which is a page can form no losing nature raster printing data stream. Then, actuation of a data-processing pipeline like a data compression is chosen so that the compression ratio about each pixel of a page and the optimal combination of printing quality may be attained.

[0017] Moreover, the stream of raster print data can also be divided based on the number of bits used for expressing each of a pixel. When extreme, the pixel corresponding to the page field printed in a color can need 24 bits for expressing each of a pixel, and the pixel corresponding to the page field printed by black and white can need only 1 bit for expressing each of a pixel. By dividing the stream of raster print data based on the number of bits used for expressing each of a pixel, the compression actuation optimized about the number of bits per pixel can be used. Furthermore, the stream of the raster print data containing the pixel of color data can need color space conversion actuation to the stream of the raster print data containing the pixel of the data of binary white and black not needing color space conversion actuation.

[0018] Furthermore, division of the stream of raster print data can be due to the print resolution of a request of the page field corresponding to a pixel. For example, as for the edge (edges) of a vector graphics object (objects), or the edge of an image object, a good thing is usually obtained from high resolution by the reason of printing quality, and the contrant region of a vector graphics object or the contrant region of an image object can be printed with a low resolution, without sacrificing printing quality. Dividing a raster printing data stream based on resolution can enable reduction of the amount of data required for processing of a data-processing pipeline, and time amount required for this to print a page can be decreased. Furthermore, halftone and compression actuation can be optimized about each of

the stream by which raster print data were divided based on the resolution of the raster printing data stream by which specification was divided.

[0019] A data-processing pipeline contains the 1st pipeline processing units, such as no losing nature compression / stretcher, a halftone unit, color space converters, or those combination. The 1st pipeline processing unit receives the 1st thing of two or more data streams including the 1st input. The 1st pipeline processing unit is constituted so that the 1st conversion called the combination of compression actuation, elongation actuation, color space conversion, halftone actuation, or the above-mentioned actuation may be performed about the data element of the 1st data stream among two or more data streams, and it generates the data stream from which the 1st was changed.

[0020] Furthermore, a data-processing pipeline contains the 2nd pipeline processing units, such as loss nature compression / stretcher, a halftone unit, color space converters, or those combination. The 2nd pipeline processing unit receives the 2nd data stream of two or more data streams including the 2nd input. The 2nd pipeline processing unit is constituted so that the 2nd conversion called the combination of the actuation compression actuation and mentioned [which mentioned above and halftone-operated / which operated it and color-space-operated / which operated it and elongation-operated] above may be performed about the data element of the 2nd data stream of two or more data streams, and it generates the data stream from which the 2nd was changed.

[0021] Furthermore, a data-processing pipeline contains the merge unit constituted so that the data stream from which the data stream from which the 1st was changed, and the 2nd were changed might be received. A merge unit is constituted so that the data stream from which the data stream from which the 1st was changed, and the 2nd were changed may be merged to an output data stream using a merge data stream.

[0022]

[Embodiment of the Invention] This invention is not limited to the gestalt of the typical operation illustrated here. Furthermore, although a print-data processing pipeline's operation gestalt describes a color laser printer In order to attain performance advantageously by using for him the juxtaposition pass which deals with the data alternatively compressed using no losing nature or the loss nature data compression approach after understanding these contents of an indication to this contractor of the technical field concerned, Probably, it turns out that a print-data processing pipeline's indicated configuration can apply to other image systems. For example, the indicated print-data processing pipeline can usually apply to the lab (laboratories) of a scanner and digital photographic processing. Furthermore, the system of the arbitration for generating the image which needs memory for storage of image data can attain the optimized image quality, and can make the required amount of memory min with the application of a pipeline's indicated configuration.

[0023] While maintaining printing quality to a high level, the print-data processing pipeline who lowers memory required to keep print data is required of color printing. Using the configuration of the print-data processing pipeline who compresses print data alternatively using the compression approach of loss nature or no losing nature maintains high printing quality, and it minimizes memory required for storage of print data. Furthermore, printing quality is further improved actuation in a print-data processing pipeline by Lycium chinense in a suitable location, and the required amount of memory is decreased.

[0024] Drawing 1 is a block diagram showing general implementation of the data-processing pipeline 200 for processing two or more data streams generated from the partition of the source-data stream formed from two or more data elements. Each of two or more data streams is formed from a data element. Although only the 1st data stream 201 and 2nd data stream 202 of two or more data streams are not expressed to drawing 1, depending on the property of a data element, from 2, a source-data stream can be divided into many data streams, and can process a data element the optimal.

[0025] The 1st pipeline processing unit 203 is used for performing 1st conversion about the 1st data stream 201 among two or more data streams. The 1st conversion performed by the 1st pipeline processing unit 203 can perform 1 or two or more actuation like a data compression or elongation, color space conversion, halftone processing, or other data-processing actuation. The method of performing 1st conversion is optimized by the property of the data element which forms the 1st data stream 201 of two or more data streams. The 1st pipeline processing unit 203 generates the data stream 205 from which the 1st was changed.

[0026] The 2nd pipeline processing unit 204 is used for performing 2nd conversion about the 2nd data stream 202 among two or more data streams. The 2nd conversion performed by the 2nd pipeline processing unit 204 can perform 1 or two or more actuation like a data compression or elongation, color space conversion, halftone processing, or other data-processing actuation. The method of performing 2nd conversion is optimized by the property of the data element which forms the 2nd data stream 202 of two or more data streams. The 2nd pipeline processing unit 204 generates the data stream 206 from which the 2nd was changed.

[0027] The merge unit 207 receives the data stream 206 from which the data stream 205 from which the 1st was changed, and the 2nd were changed. The data stream 206 from which the data stream 205 from which the 1st was changed, and the 2nd were changed is merged to the output data stream 208 by the merge unit 207. In order to perform this merge actuation, the merge unit 207 uses the merge data stream 209. Since the merge data stream 209 is formed from a merge data element and it is set by the 1st data stream 201 and 2nd data stream 202 of two or more data streams, the data element of the output data stream 208 which it is as a result of merge actuation has the sequence of the data element in a source-data stream. Therefore, the merge unit 207 assembles the data stream 206 from which the data stream 205 from which the 1st was changed, and the 2nd were changed, and holds the sequence of the data element in a source-data stream to the output data stream 208.

[0028] By dividing a source-data stream and forming the 1st the 201 of two or more data streams, and 2nd 202 based on the property of the data element in a source-data stream, the 1st pipeline processing unit 203 and the 2nd pipeline processing unit 204 are constituted, respectively so that processing of a data element may be optimized.

[0029] Drawing 2 is the simplified conceptual diagram showing the flow of the print data which pass along a color printer, and includes the desirable operation gestalt of the print-data processing pipeline 15 about a color laser printer. Drawing 2 uses the pass of the loss nature of juxtaposition, and no losing nature, and tries instantiation of only the flow of the print data which pass along the typical actuation performed by the print-data processing pipeline 15. Drawing 2 is not the print-data processing pipeline's 15 hardware block diagram. The print-data processing pipeline 15 can consist of various different approaches so that processing of print data may be optimized, so that it may become clear from the

explanation after a hardware block diagram.

[0030] The print-data input to a system is sent in the format that some differ. Input print data are sent in the form of a display list (Display List), raster print data, or the raster print data that already received compression. Display list print data include information required to constitute the page which should be printed. Display list print data can contain raster print data together with a text or the raster print data as image print data, the code of the printing control language showing a text alphabetic character, the code of the graphics language showing a graphics image, or the code that identifies some combination of these print data. Notionally, print data are inputted into the print-data processing pipeline 15 in the suitable location depending on the processing needed for a format required to generate a printout changing print data.

[0031] An image processing is performed about display list print data in the image-processing actuation 1 by activation of a firmware routine. Dividing an input into the piece of a page (page strips) is included in this image-processing actuation 1. In order to divide an input into the piece of a page, display list print data are kept based on the vertical position of the page to which print data correspond. It follows taking into consideration the class of print data which form the section of the page corresponding to the piece of a page formed on dividing a page into the piece of a page by the image-processing actuation 1. Each piece of a page is formed with the piece element of a page of 1 or 2. If the section of a page is completely formed in loss nature or completeness from one of the print data of no losing nature, only the piece of a page of one element will be formed by the image-processing actuation 1 about the section of a page. However, when the display list print data about the section contain both print data of loss nature and no losing nature, two separate piece elements of a page are formed about the section of the page. In another side, one side of these piece elements of a page contains only no losing nature print data only including loss nature print data.

[0032] Formation of the piece element of a page of the corresponding loss nature about the section of a page and no losing nature is generated when both the classes of data exist in display list print data. In this case, the image-processing actuation 1 makes each of two piece elements of a page by filling appropriately the piece element of a page of both loss nature and no losing nature with the piece element of a page of the loss nature from a page section, and no losing nature the whole pixel location. The print data about the section of a page are contained in the piece element of a page of loss nature and no losing nature. For example, when display list print data contain no losing nature text print data together with loss nature image print data, as for the image-processing actuation 1, display list print data are divided into text print data and image print data. Then, each of the piece element of a page of no losing nature and loss nature is filled with each text print data and image print data for every pixel location. In this case, it is considered that the piece of a page corresponding to display list print data is a piece of a synthetic page containing the piece element of a page which includes the loss nature image print data put with the piece element of a page which includes no losing nature text print data.

[0033] When the piece element of a page of loss nature and no losing nature is formed, the merge plane (merge plane) to which the image-processing actuation 1 corresponds is generated. This merge plane consists of 1 bit about each pair of the pixel location which corresponds in two piece elements of a page, and points out the piece element of a page of the loss nature in which the print data about that pixel are contained, and no losing nature. The bit which forms a merge plane is formed by the cutting tool, and is

transmitted through the print-data processing pipeline 15 as byte stream. This merge plane is used in the phase after the print-data processing pipeline 15, the piece element of a page of no losing nature and loss nature is combined, and, thereby, the first piece of a page (it is original) is reconfigured. This merge actuation is stated to a detail in the direction behind this specification.

[0034] When one piece element of a page is formed so that it may have only loss nature print data or no losing nature print data, all the print data about the piece of a page are contained in one piece element of a page. Furthermore, the merge plane relevant to the piece element of a page is not generated depending on the image-processing actuation 1. Since there is only one piece element of a page about the section of a page, the bit of a merge plain throat is also the same. Therefore, there is no need of sending the merge plane which passes along the print-data processing pipeline 15 and which consists of the cutting tool of the same value altogether. Instead of generating a merge plane by the image-processing actuation 1, a merge plane is generated in the phase after the print-data processing pipeline 15. This is described in detail in the direction behind this specification.

[0035] Moreover, the image-processing actuation 1 generates the halftone plane (halftone plane) which has one value corresponding to each pixel location in the piece of a page. The value of each halftone is expressed with 2 bits, and it opts for the halftone actuation applied to the print data corresponding to a pixel so that this specification may describe later. When the halftone actuation performed is the same about each pixel of the piece of a page, a halftone plane is generated in the phase in the direction of behind the print-data processing pipeline 15. This is stated more to a detail in the direction behind this specification.

[0036] From the display list print data divided into the piece element of a page, raster print data are generated corresponding to display list print data. The piece element of a page of the raster print data of a result passes along actuation of the print-data processing pipeline 15, when generated. The raster printing data memory 2 is used for keeping the raster print data generated by the image-processing actuation 1. The memory area assigned to the raster printing data memory 2 has sufficient storage capacity to include the raster print data of two piece elements of a page with the number of Rhine in which the maximum of per the piece of a page is possible, the size of a corresponding merge plane, and the size of a corresponding halftone plane. Raster processing is carried out in the image-processing actuation 1, and each of the piece element of a page is kept in the memory area assigned to the raster printing data memory 2, is sent to the next actuation of the print-data processing pipeline 15 after that, and makes a field to the raster printing data memory 2 for the following piece element of a page by which raster processing was carried out by the image-processing actuation 1. In the desirable operation gestalt, the raster printing data memory 2 should be cautious of not being contained in the integrated circuit with which the print-data processing pipeline 15 is mounted.

[0037] Usually, the generated raster print data consist of three 8-bit cutting tools about each pixel. It corresponds to one of the color dimensions (color dimensions) in the color space of 3 bytes as which display list print data are expressed, respectively. In the color printer by which the gestalt of desirable operation of a print-data processing pipeline operates, this is RGB color space. To this contractor of the technical field concerned, however, the color space of input print data Cyanogen, a Magenta, yellow, black (CMYK), and a hue (hue), a saturation ratio (saturation), Lightness (value) (HSV), a hue, lightness (lightness), A saturation ratio (HLS), brightness (luminance), a **** scale, a copper rust scale

(La*b*), Probably, it turns out that they are brightness, a red-green scale, a yellow-blue scale (Luv), brightness, a **** scale, a greenish yellow scale (YCrCb), or the color space of arbitration like YIQ. The generated raster print data correspond to an image, a text, a line drawing, graphics, or some of such combination.

[0038] As mentioned above, the image-processing actuation 1 creates a merge plane, when it is display list print data with which the piece element of a page of no losing nature and loss nature is formed. In the direction behind a print-data processing pipeline, a merge bit is used for choosing a pixel from the pair of the piece element of a page of no corresponding losing nature and loss nature, combines the piece element of a page of no losing nature and loss nature, and maintains the exact spatial relations between pixels towards the first image (it is original).

[0039] Color space conversion is performed by the color space conversion actuation 3. The degree of the compression attained about raster print data is influenced by the color space where raster print data are expressed. For example, usually it is finished by a high loss nature compression ratio's performing color space conversion to YCbCr color space from RGB color space (in the same image quality), and performing loss nature compression actuation, after changing into YCbCr. YCbCr color space is the color space of brightness / hue / saturation (chroma) mold, and Cr and Cb element of color space include the information on both a hue and saturation, respectively. Human being's eye is the most sensitive to modification of brightness, and is not relatively sensitive to modification of saturation. For this reason, the print data expressed in this color space contain the amount of remarkable redundancy in Cr and Cb element. Loss nature compression is relatively performed about the piece element of a loss nature page of raster print data with a large compression ratio as a result. The advantageous point of this color space depending on a compression ratio does not exist about no losing nature compression. The reason is that the piece element of no losing nature page does not receive the color space conversion to YCbCr color space from RGB color space. However, when this color space conversion was needed by other reasons, it performs. In the phase after a print-data processing pipeline, both the piece element of no losing nature page and the piece element of a loss nature page receive the color space conversion in a CMYK color specification from RGB or YCbCr color space. This color space conversion is described in the direction behind this specification.

[0040] Forming the piece of a page of no losing nature and loss nature from the section of the page corresponding to one piece of a page, and generating a merge plane and a halftone plane further makes the amount of data sent through the print-data processing pipeline 15 increase. The merge bit and two halftone bits relevant to each pixel make the amount of print data increase. However, a memory area required to include print data since a high compression ratio is performed about the piece of a page of no losing nature and loss nature and a merge plane and a halftone plane moreover also receive no losing nature compression only increases slightly by the case of being the worst. However, the amelioration result of the printing quality through the pipeline who does not apply the algorithm of a data compression which is different to the data of no losing nature and loss nature is remarkable. for almost all print jobs -- print data -- all -- loss nature -- or it is one of no losing nature altogether. The only increment in the amount of the raster print data sent through the print-data processing pipeline 15 in these cases from the image-processing actuation 1 is a halftone plane when applying a halftone algorithm which is different in the pixel of the piece of a page. When the same halftone algorithm is

applied to all the pixels of the piece of a page, in the image-processing actuation 1, a halftone plane is not generated at all.

[0041] The direct input of the raster print data can be carried out to a print-data processing pipeline. Raster print data can be sent from a scanner or a host computer. Moreover, the equipment which offers raster print data offers the header used by the image-processing actuation 1, and forms the piece of a page. The information which the image-processing actuation 1 uses for separating raster print data as loss nature or no losing nature for the image-processing actuation 1 is included in this header.

[0042] Before the piece element of a page of loss nature raster print data receives loss nature compression, Cr of the piece element of a page and Cb element with which color space conversion of the raster print data was carried out are decreased alternatively, the amount of the print data which the print-data processing pipeline 15 has to deal with is decreased, the amount of data processing is made to increase by this, and the amount of memory required to hold raster print data is decreased. The redundancy of Cr and Cb element which were mentioned above makes it possible to attain subsampling of these raster print data, and the property of no visual losing nature of raster print data is maintained. Subsampling actuation is performed by the loss nature compression actuation 5. The loss nature compression actuation 5 compresses along with the scanning line about the block of the raster print data formed from the section which is 8-pixel width of face and is the height for the scanning line which 8 followed. It passes along subsampling and these 8 pixel x 8 pixel numbers of the whole blocks decrease. [0043] The amount of subsampling to generate enables it to suit the memory area where raster print data are usable depending on a required data decrement. If sufficient memory area is usable, subsampling will be performed so that the amount of the loss nature raster print data of the piece element of a page may be lowered only one third. When the loss nature raster print data to memory need to be suited, subsampling is performed so that only one half may lower the amount of the loss nature raster print data of the piece element of a page.

[0044] Subsampling which attains one third of reduction is performed as follows. For example, it thinks of three groups of loss nature raster print data, and suppose that each group expresses four blocks of a pixel of 8x8 from one color space. The 1st group is formed from a brightness element, the 2nd group is formed from Cr element, and the 3rd group is formed from Cb element. And there are 12 blocks of loss nature raster print data as a whole. Into the 2nd group, only 4 blocks only of loss nature raster print data corresponding to the mutual pixel which met each scanning line are maintained about each of the scanning line of 8 in each (it starts in the 1st pixel of each scanning line of each block). As a result, 8-pixel width of face decreases to 4 pixels about each scanning line of each by 4 blocks. By this approach, subsampling of the 4 blocks of Cr element is carried out to 2 blocks. Subsampling of Cb element is the same way and becomes 2 blocks from 4 blocks. After this subsampling, eight of 12 blocks of the beginning of a pixel remain as they are, namely, they decrease only by one third.

[0045] Subsampling which attains reduction in half is made by applying the same subsampling procedure as decreasing 1/3 in which 2 blocks of Cr element and 2 blocks of Cb element remain. If the same subsampling procedure is applied to the remaining block of Cr and Cb element, the result of 1 block will be produced [element / 1 block and / Cb] about Cr element. Six remain among the first 12 blocks of a pixel after this 2nd pass of subsampling, and only one half decreases in number.

[0046] After subsampling actuation is performed, in the loss nature compression actuation 5, a loss

nature compression algorithm is applied to the piece element of a page. The loss nature compression approach used in the print-data processing pipeline's 15 desirable operation gestalt is a well-known JPEG algorithm. The JPEG loss nature compression approach was chosen because the hardware developed before used easily [performing this approach]. However, the loss nature compression approach of arbitration like vector quantization has been used. When it is a piece element of no losing nature page, compression of this piece element of a page is performed by no losing nature compression actuation 4. Furthermore, when there are the merge and halftone plane corresponding to the piece element of a page, these both are compressed by no losing nature compression actuation 4. the print-data processing pipeline's 15 desirable operation gestalt -- a jib - Lempel's no losing nature compression / elongation approach is used, and this is indicated by U.S. Pat. No. 5455576, and is taken in by reference here. However, JBIG, a run length, or the no losing nature compression / elongation approach of arbitration like delta low compression (deltarow compression) can also be used. The information about the technique used for carrying out no losing nature and loss nature compression is "INTRODUCTION TO DATA COMPRESSION" and Khalid. Sayood, 1996, Morgan A book called Kaufmann publication sees and it takes in by reference here.

[0047] The compression raster print data of the loss nature generated by the loss nature compression actuation 5 and no losing nature compression actuation 4, respectively and no losing nature are ** kept by the compression raster printing data memory 6. The field assigned to the compression raster printing data memory 6 is sufficient size to hold all the planes of three colors of the raster print data of the compression loss nature of a whole page, and no losing nature, the compression halftone data of a whole page, and the compression merge data of a whole page. In the desirable operation gestalt, the compression raster printing data memory 6 should be cautious of not being contained in the integrated circuit with which the print-data processing pipeline 15 is mounted. Furthermore, assignment of the raster printing data memory 2 and the compression raster printing data memory 6 is the notional semantics only for the purpose explaining actuation of the print-data processing pipeline 15. If it will be about the raster print data with which compression/elongation actuation was not performed when both the raster printing data memory 2 and the compression raster printing data memory 6 were probably compression raster print data, it will be physically put on a system memory and print data will be kept. Memory is that which is said if only it specifies according to the class of data kept by given time amount, and the location of a system memory can be used for keeping the class of both print data of different time amount.

[0048] Raster print data are expressed with 3 bytes about each pixel of the piece element of a page of loss nature or no losing nature, namely, it is expressed with 1 byte about the element of each color space of raster print data. The piece element of a page of loss nature and no losing nature moves two parallel channels through the print-data processing pipeline 15. By the location of the print data in the print-data processing pipeline 15, the raster printing data stream of loss nature and no losing nature is either 1-byte width of face or 3-byte width of face. The raster print data of loss nature and no losing nature start the color space conversion actuation 3 and 9 as a 3-byte width-of-face stream, come out as a 1-byte width-of-face stream, and are packed to the 3-byte width-of-face stream which has each cutting tool of the 3-byte group showing the dimension of output color space. The raster print data of loss nature and no losing nature start the color space conversion actuation 12 as a 3-byte width-of-face stream, and

come out as a 1-byte width-of-face stream. Each cutting tool of the 3-byte group included in the color space conversion actuation 3, 9, and 12 expresses the raster print data corresponding to one dimension of input color space. 1 byte as a result which comes out of the color space conversion actuation 12 expresses the raster print data corresponding to one dimension of the output color space which has received printing. The print data of loss nature and no losing nature come out of the image-processing actuation 1 as a 3-byte width-of-face stream. The print data of loss nature and no losing nature start the loss nature 5 and 8, 11 compression and no losing nature 4 and 7, and 10 compression actuation as a 3-byte width-of-face stream, come out as a 1-byte width-of-face stream, and a buffer is carried out after that, and they serve as a 3-byte width-of-face stream. The print data of loss nature and no losing nature start the loss nature 5 and 8, 11 elongation and no losing nature 4 and 7, and 10 elongation actuation as a 1-byte width-of-face stream, and come out as a 1-byte width-of-face stream. The raster print data of loss nature and no losing nature enter as a 1-byte width-of-face stream, and come out of both the merge actuation 13 and the halftone actuation 14 as a 1-byte width-of-face stream.

[0049] A merge data plane is adjusted to a cutting tool. Since only 1 bit is needed for merge actuation about each pixel, each cutting tool of merge data includes the merge information about 8 pixels. A halftone data plane is adjusted to a 8-bit cutting tool. Since 2 bits is needed by halftone actuation about each pixel, each cutting tool of halftone data includes the halftone information about 4 pixels. Both a halftone plane and a merge plane move two juxtaposition cutting tool width-of-face channels through a print-data processing pipeline.

[0050] The optimal combination of memory compaction and image quality is attained about print data by forming the piece element of a page of loss nature and no losing nature, and compressing the raster print data of these piece elements of a page using the compression approach of loss nature or no losing nature, respectively after that. Loss nature raster print data are high-compressed using loss nature compression technology, and maintain the property of no losing nature visually. Since no losing nature raster print data are compressed using no losing nature compression technology, degradation of image quality is not produced but a high compression ratio is attained. The compression approach of the selected loss nature and no losing nature is optimized about loss nature raster print data (image) and no losing nature raster print data (a text, a line drawing, graphics), and a high compression ratio is produced, without degrading image quality. This function is a very advantageous point and it is generated from using the piece element of a page of loss nature and no losing nature in the print-data processing pipeline 15. Although two or more channels are used and the raster print data of loss nature and no losing nature, corresponding halftone, and merge data are moved through the print-data processing pipeline 15 with the print-data processing pipeline's 15 desirable operation gestalt, it should be cautious of the ability of the piece element of a page of print data for one multiplexer channel to be used and to also be transmitted continuously. In order to reduce required hardware, performance is sacrificed for use of one multiplexer channel.

[0051] It is possible to input compression raster print data into the print-data processing pipeline 15. This can be produced when compression raster print data are the inputs of the print-data processing pipeline 15 relevant to the command of a display list, and raster print data or compression raster print data can be a print-data processing pipeline's input in preparation for elongation and printing of a whole page. These compression raster print data can be given with the scanner through a host computer or a

host computer. The header attached with a host computer including the information used by the image-processing actuation 1 is contained in these compression raster print data.

[0052] The case where he hopes to put the input before the loss nature image to the print-data processing pipeline 15 as display list print data in the loss nature image which is an input to the print-data processing pipeline 15 as compression loss nature raster print data (overlay) is considered. The print-data processing pipeline 15 is provided with pass. Compression raster print data from the compression raster printing data memory 6 to the loss nature elongation actuation 8 In this case, delivery, Color space conversion is performed by the color space conversion actuation 9 (when compression loss nature raster print data are not expressed in RGB color space). It returns to RGB color space and a result is kept to the raster printing data memory 2, and in order to perform covering using a pre- input disadvantage degeneration image, elongation loss nature raster print data are distributed to the image-processing actuation 1. The image-processing actuation 1 performs division into the piece element of a loss nature page after that. This pass for combining compression loss nature raster print data with no losing nature print data is shown to drawing 2 by notional semantics. Behind this specification describes the actual root which performs actuation by which print data follow functional block of hardware and are shown in drawing 2 .

[0053] As mentioned above, the print-data processing pipeline 15 can receive a raster or compression raster print data from various equipments like the host computer which can supply printing of a raster or a compression raster print-data format, or a scanner. Receiving either a raster or compression raster print data removes the need for the image-processing actuation 1 of performing raster actuation. furthermore, when print data are received as compression raster print data whose printing is already possible, there is no need of performing the loss nature 5 and (or) no losing nature 4 compression actuation, or color space conversion actuation 3.

[0054] Functional block used for attaining the actuation shown in the print-data processing pipeline's 15 desirable operation gestalt is realized by one integrated circuit. The technique used for realizing these functions with a digital integrated circuit is common knowledge in the design field of a digital integrated circuit. Implementation of the print-data processing pipeline's 15 function in exclusive hardware offers the processing performance which was excellent about the print data included in a pipeline.

[0055] Although it is possible to perform each of the actuation in the print-data processing pipeline 15 using a microprocessor, the processing performance of print data falls extremely. Although a microprocessor performs required print-data compression, color space conversion, and print-data elongation actuation, it is necessary to deal with the piece element of a page of loss nature and no losing nature by turns. Furthermore, the piece element of a page of loss nature and no losing nature is merged by the microprocessor, and it performs halftone actuation.

[0056] The print-data processing pipeline's 15 desirable operation gestalt operates using the color laser print engine 16 designed so that the print data expressed by the CMYK color specification may be received. Furthermore, the print engine 16 operates from developing a toner continuously about each plane of a CMYK color specification (developing), before transmitting a toner to a page. How the raster print data with which the development of four pass used with the print engine 16 passes along the print-data processing pipeline 15 flow is ordered. The raster print data about a page are sent to the print

engine 16 with four continuous pass by this mode of operation of the laser print engine 16 (each dimension of a CMYK color specification one pass).

[0057] As mentioned above, the compression raster print data of a whole page are accumulated by the memory assigned to the compression raster printing data memory 6 when the piece element of a page of loss nature and no losing nature is compressed. Loss nature compression raster print data are expressed in YCbCr color space. No losing nature compression raster print data are expressed in RGB color space. In order to change compression raster print data into each of four planes of the CMYK color specification of a laser print engine, it is necessary to pass along the print-data processing pipeline 15 by four continuous pass, and to send the compression raster print data about a page. In each of these 4 pass, the compression raster print data of no losing nature and loss nature are sent to each of no losing nature 10 and loss nature 11 elongation actuation through two 1-byte width-of-face channels for elongation. A compression merge plane and a halftone plane are interleaved by the cutting tool of no losing nature compression raster print data (interleave, insertion). Since a merge plane and a halftone plane are interleaved by no losing nature raster print data, these receive no losing nature compression together with no losing nature raster print data. A merge plane and a halftone plane are extracted from no elongation losing nature printing data stream after elongation. This is described in detail in the direction behind this specification. Loss nature compression print data contain the piece element of a loss nature page. Interpolation actuation is performed after elongation of loss nature compression raster print data, and the value of Cr removed through subsampling in advance of compression of loss nature raster print data and Cb element is extended (expand). Interpolation actuation is performed as a part of loss nature elongation actuation 11.

[0058] Both no losing nature print data and loss nature print data are sent for color space conversion of the color space conversion actuation 12 after elongation actuation of the loss nature 11 and no losing nature 10. Since the print engine 16 uses the development of four pass, the compression raster print data about all the color planes (color plane) of a page are continuously sent 4 times through the loss nature 11 and no losing nature 10 elongation actuation. About each of four pass to the color space conversion actuation 12 of the compression raster print data which pass along elongation actuation of no losing nature 10 and the loss nature 11, the color space conversion actuation 12 generates one of four planes of a CMYK color specification. Color space conversion actuation 12 is performed using three 8-bit cutting tools of the raster print data of the loss nature showing each of the pixel of the page which should be printed, or no losing nature. The raster printing data stream output from the color space conversion actuation 12 contains two 8-bit channels formed from loss nature raster print data and no losing nature raster print data. No losing nature raster print-data input to the color space conversion actuation 12 is expressed in RGB color space. The loss nature raster print-data input to the color space conversion actuation 12 is expressed in YCrCb color space.

[0059] Drawing 2 suggests that the loss nature 5 and no losing nature 4 compression actuation, no losing nature 7 and loss nature 8 elongation actuation, no losing nature 10, and loss nature 11 elongation actuation are attained in a separate hardware unit. However, in the print-data processing pipeline's 15 desirable operation gestalt, there is a hardware unit which performs compression and elongation of loss nature, and there is a hardware unit which performs compression and elongation of no losing nature. These units perform all compression/elongation actuation about the print-data processing pipeline 15.

Although the color space conversion actuation 3, 9, and 12 is shown as a hardware unit with separate drawing 2 in a desirable operation gestalt for the purpose explaining the print-data processing pipeline 15, one color space converter is used for performing all the required color space conversion. In the print-data processing pipeline's 15 desirable operation gestalt, the color space converter is indicated by application "APPRATUS FOR GENERATING INTERPOLATION INPUT DATA" (10960608-1) and "APPRATUS FORROUTING INTERPOLATION INPUT DATA" (10960467-1), and takes in by reference here. Probably, this contractor of the technical field concerned understands the color specification used with the color space transducer which performs conversion between the color space chosen as expressing the print data of loss nature and no losing nature, and the laser print engine suitable for using it on two or more pass print-data processing pipeline, after understanding this specification.

[0060] The merge actuation 13 receives no losing nature raster printing data stream, a loss nature raster printing data stream, a halftone data stream, and a merge data stream as an input. Each of these printing data streams is formed from byte stream. The print data which express one color plane of the color specification of a print engine about each of the pixel of a whole page are contained in the raster printing data stream of no losing nature and loss nature. The function of the merge actuation 13 combines these two raster printing data streams for every pixel, and reconfigures the first image (it is original) before divided into the piece element of a page of no losing nature and loss nature about the color plane currently processed. It judges whether the bit of the merge data relevant to each pixel is used by the merge actuation 13, and the raster print data of the pixel by which current processing is carried out are chosen from no losing nature raster printing data stream, or are chosen from a loss nature raster printing data stream. The following pass with which the raster print data of the loss nature which passes along the merge actuation 13, and no losing nature happen successingly reconfigures the first image about the color plane with which the color specification of a print engine remains. It corresponds to the location on the page by which the pixel was printed, respectively. It is considered that reconstruction processing is the spatial synchronization of the raster printing data stream of no losing nature and loss nature, and it generates the first image. Furthermore, in order to attain a spatial synchronization, the raster printing data stream of no losing nature and loss nature and the stream of merge data must synchronize in time.

[0061] The indicated print-data processing pipeline 15 uses the development property of the print engine 16 for every plane, and lowers the amount required to carry out printing actuation of the whole memory. In the print-data processing pipeline 15, there is only the need of saying that the merge actuation 13 uses the line buffer of the minimum size, about one color plane of the CMYK color specification of the print engine 16 for storage of raster print data by positioning the merge actuation 13 in the degree of the color space conversion actuation 12. The need of saying that the merge actuation 13 uses the line buffer of size sufficient in the merge actuation 13 having been located in the print-data processing pipeline 15 in advance of the color space conversion actuation 12 to keep raster print data about each of the three dimension of RGB and YCrCb color space will arise.

[0062] The configuration of the print-data processing pipeline who uses pass separate about no losing nature and loss nature print data has the print engine and compatibility which operate by receiving raster print data about all the color planes of the color specification of a print engine with one pass instead of.

[two or more pass] In this case, the color space conversion actuation 12 needs to perform color space conversion about the raster print data showing each pixel, and needs to generate each of the value corresponding to coincidence in C, M, Y, and K element. This needs the color space conversion actuation 12 of having the function to perform interpolation required to generate C, M, Y, and each K element to coincidence. The line buffer of sufficient size to include raster print data about C, M, Y, and each K element is still more nearly required for the merge actuation 13.

[0063] With the print-data processing pipeline's 15 desirable operation gestalt, the raster print data together put by the merge actuation 13 are given to the halftone actuation 14. The halftone actuation 14 is positioned so that it may become actuation of the last performed about raster print data in advance of distribution of the print data to the print engine 16. The halftone actuation 14 performs halftone processing for every plane about the merged raster print data, as shown by the halftone print data relevant to each pixel. Positioning the halftone actuation 14 after the merge actuation 13 makes it possible to process continuously each color plane of the color space where the halftone actuation 4 is used with the print engine 16, and fewer memory is sufficient for keeping raster print data by this. Furthermore, since there is the need of performing halftone actuation 14 separately about the stream of no losing nature raster print data and loss nature raster print data when halftone actuation 14 is performed in advance of merge actuation, the complexity of the halftone actuation 14 increases remarkably. Furthermore, putting the halftone actuation 14 on the print-data processing pipeline's 15 last brings about the optimal printing quality. Positioning the halftone actuation 14 in beforehand [of compression of loss nature raster print data] brings about some loss of the emphasis (enhancements, enhancement) attained by halftone processing, and it decreases the compression level attained about halftone raster print data. In addition, the record level of printing quality is attained by performing halftone actuation 14 about the raster print data expressed with the color specification of a print engine. Therefore, with the print-data processing pipeline's 15 desirable operation gestalt, the halftone actuation 14 is located after the color space conversion actuation 12.

[0064] Assigning a halftone bit to each pixel of raster print data is performed by the image-processing actuation 1 which makes a halftone plane. As mentioned above, the print data included in a print-data processing pipeline can be the text of a raster print-data format, graphics, image print data, the print data of printing control language, and the print data of graphics language. Firmware actuation of the image-processing blocked operation 1 generates a halftone plane about the piece element of a page of the loss nature which corresponds according to the class of halftone algorithm applied, and no losing nature. For example, the halftone bit about text print data can be assigned to "11", the halftone bit about graphics print data can be assigned to "10", the halftone bit about image print data can be assigned to "01", and the halftone bit which shows that any halftone actuation is not applied to the print data corresponding to a pixel can be assigned to "00." Assignment of a halftone bit enables application of the halftone actuation optimized about the class of print data relevant to a halftone bit. for example, a halftone bit -- per inch -- ** -- the halftone algorithm which has the Rhine screen (line screen) of the number of Rhine can be specified. Halftone actuation is common knowledge in a printing technical field. The further information about halftone can be seen in ISBN0-262-21009-6(it will be 4th edition in 1993) Ulichney, and "Digital Halftoning" of R, and is taken in by reference here.

[0065] The merge actuation 13 receives no losing nature raster printing data stream, a loss nature raster

printing data stream, a halftone printing data stream, and a merge printing data stream as an input. Each of these printing data streams is formed from a 8-bit cutting tool. The print data which express one color plane of a print engine color specification about each of the pixel of a whole page are included by the raster printing data stream of no losing nature and loss nature. The function of the merge actuation 13 is combining these two raster printing data streams for every pixel, as the first image (it separated into the piece element of a page of no losing nature and loss nature before) is reconfigured about the color plane currently processed. The following pass which the raster print data of the loss nature which passes along the merge actuation 13, and no losing nature start successingly reconfigurates the first image about the plane in which the print engine color specification remains. It corresponds to the location on the page of a pixel printed, respectively. It is considered that reconstruction processing is the spatial synchronization of the raster print data of no losing nature and loss nature, and it generates the first image. Furthermore, in order to attain a spatial synchronization, the raster printing data stream of no losing nature and loss nature and the printing data stream of halftone must synchronize in time.

[0066] The print-data processing pipeline's 15 configuration is performed before passage of the print data which pass along the print-data processing pipeline 15, and in the meantime. Some a pipeline's parameters are constituted by each page, some of each color planes, and some of each pieces of a page. The number of Rhine per page and the number of pixels around Rhine can be constituted for every page. Furthermore, the number of the color planes processed by the printing processing pipeline 15 (that is, it is full color and they are monochrome or the image printed) can constitute for every page.

[0067] Processing the raster print data about each color plane needs the configuration of the color space conversion actuation 3, 9, and 12. In advance of activation of the color space conversion actuation 3, 9, and 12, a color space translation table required for conversion to the color plane which output color space meant is loaded. Furthermore, processing of each color plane which passes along the halftone actuation 14 needs the configuration of the halftone actuation 14 so that it may be used, in order that a suitable halftone table may carry out halftone processing of the color plane. For example, when processing the continuous color plane of an image, each halftone table can respond to the halftone algorithm which uses the Rhine screen (line screen) chosen in a different relative viewpoint. Use of the Rhine screen chosen in a relative viewpoint which is [about the color plane printed continuously] different decreases the interaction between the color planes which generate printing processing so that **** [the technical field of halftone].

[0068] Moreover, the print-data processing pipeline 15 has the parameter which can be constituted for every piece of a page. The image-processing actuation 1 can constitute the print-data processing pipeline 15, and performs various usable pipeline processing actuation about each piece of a page. For example, about every piece of a page, the image-processing actuation 1 can constitute the print-data processing actuation 15, and can perform or bypass the compression actuation 4 and 5, the elongation actuation 7, 8, 10, and 11, the color space conversion actuation 3, 9, and 12, or the halftone actuation 14. Furthermore, 1, 2, 4, and 8 can be specified for the number of bits per pixel of the piece element of no losing nature page for every page. With the print-data processing pipeline's 15 desirable operation gestalt, the compression algorithm used by the loss nature compression actuation 5 needs 8 bits per pixel of the piece element of a loss nature page. furthermore, the image-processing actuation 1 -- the loss nature of the piece of a page, or no losing nature element -- null -- it can specify that it can form

from data (null data), and can specify that each bit of a corresponding merge plane is the same value. [0069] Drawing 3 is the simplified hardware block diagram in the print-data processing pipeline's 15 desirable operation gestalt. The pass connected to the functional block diagram shown in drawing 3 expresses a data path. The various control lines used for controlling the data flow between functional block are not shown in drawing 3. With a desirable operation gestalt, compression/elongation actuation 4, 5, 7, 8, 10, and 11, the color space conversion actuation 3, 9, and 12, the merge actuation 13, and halftone actuation 14 are performed by ASIC120. The print-data processing pipeline's 15 desirable operation gestalt is constituted so that ASIC120 may be connected to a PCI bus. However, probably, it turns out that this contractor of the technical field concerned can constitute so that the print-data processing pipeline 15 may be connected to the bus of other classes like a VESA bus. A PCI bus architecture is common knowledge in the technical field of digital system design, and is not stated to a detail here. Image-processing actuation 1 is performed by processor like a microprocessor 132, and communicates with the print-data processing pipeline 15 through a PCI bus. At the time of the transfer to ASIC120, and the transfer from ASIC120, print data are searched from the raster printing data memory 2 and the compression raster printing data memory 6, and are loaded to memory 2 and 6.

[0070] ASIC120 connects with a PCI bus through the PCI bus interface 121. The PCI bus interface 121 contains a register required for buffering of print data, when flowing between a register and a PCI bus required for the configuration of the PCI bus interface 121, and ASIC120.

[0071] A Direct-Memory-Access controller like video DMA 122 controls the flow of some print data of ASIC120 which performs the print-data processing pipeline's 15 various functions, and the flow of the print data from a part of ASIC120. The video DMA buffer 123 fits the printing data rate between a PCI bus and various functional block of ASIC120, when passing video DMA 122, and print data are kept temporarily, and print data pass ASIC120 and it moves from ASIC120. Use of a video DMA buffer compensates the PCI bus interface 121 and the mismatching in the rate of the data flow between functional block in the remaining part of ASIC120.

[0072] Some important advantageous points arise from the implementation of the print-data processing pipeline 15 in ASIC120 as a result. The 1st advantageous point is the efficient method that the processing actuation about the print-data processing pipeline 15 is attained inside ASIC120. Although print data are between the actuation which continued without inviting the overhead of processing and are transmitted by putting all actuation on ASIC120, when a transfer of print data arises in the PCI bus between integrated circuits with separate dedication in activation of one print-data pipeline processing actuation, the overhead of processing arises. I hear that the 2nd advantageous point can constitute ASIC120, and there is so that print data may be alternatively bypassed about functional block of ASIC120. In order to transmit to other functional block, or since the 3rd advantageous point keeps it in memory, I hear that it can feed back the output of functional block to video DMA 122 alternatively, and it is. These functions are very supple, an efficient print-data processing pipeline is made, and a print-data processing pipeline is constituted so that print-data processing actuation may be optimized depending on the class of print data. Since ASIC120 is constituted so that the specific subset of possible print-data processing actuation may be performed, the actuation applied to print data suits the property of print data in the way of a broad class. This makes it possible to constitute a pipeline so that a printing system may process all the possible classes of print data the optimal. A feedback function makes it

possible to perform in the sequence which was most suitable for print-data processing actuation preparing the print data of page printing. Completion of actuation of the print-data processing pipeline 15 turns the result to the next actuation by video DMA 122.

[0073] Depending on the class of actuation of the print-data processing pipeline 15 who performs about the print data loaded from the PCI bus, video DMA 122 is constituted so that print data may be distributed to suitable functional block of ASIC120. If color space conversion is needed, print data will be sent to the piece manager 124 of a page by video DMA 122. The piece manager 124 of a page sends print data into the input of the suitable loss nature of the color space converter 125, or no losing nature from video DMA 122. Furthermore, the piece manager 124 of a page extracts the merge data and halftone data which were interleaved by no losing nature raster print data, and sends these each to the merge unit 128 through a separate cutting tool width-of-face channel. When compression or elongation of print data is required, print data are addressed to suitable compression of loss nature elongation / compressor 126, or the no losing nature compression / stretcher 127, or the input of elongation by video DMA 122, and are sent. The merge unit 128 performs merge actuation 13 about the raster printing data stream of the loss nature outputted from the color space transducer 125, and no losing nature. A line buffer 129 contains SRAM for the raster print data of the loss nature used by the merge actuation 13, and no losing nature, a merge, and buffering of halftone data. The halftone unit 130 performs halftone actuation 14 about the raster printing data stream merged corresponding to each of a pixel according to the bit specified by the halftone plane. although not clearly shown in drawing 3 , the color space transducer 125 and the halftone unit 130 are constituted so that print data may be bypassed about these functional block, when the actuation which should be performed about print data does not include color space conversion and (or) halftone processing.

[0074] When actuation is completed about the print data which pass along the print-data processing pipeline 15 of drawing 3 flowing, print data are transmitted to memory from ASIC120, and return to ASIC120 later for the further processing. For example, after compression actuation of the loss nature 5 and no losing nature 4 is performed by the compressor/stretcher of the loss nature 126 and no losing nature 127, respectively, the compression raster print data of the result are transmitted to video DMA 120 through the PCI bus interface 121, and are transmitted to the compression raster printing data memory 6 through a PCI bus after that. When compression raster print data are elongated, compression raster print data are transmitted to ASIC120 through the PCI bus interface 121 through a PCI bus from the compression raster printing data memory 6, and are sent to either the compressororstretcher of the loss nature 126 and no losing nature 127, and both elongation inputs by DMA122. after color space conversion actuation 3 is performed about the raster print data of loss nature and (or) no losing nature, in order that the raster print data of the result are returned to loss nature 126 compression / stretcher, and (or) compression/stretcher of no losing nature 127 through video DMA 122, are returned to the raster printing data memory 2 through video DMA 122 and the PCI bus interface 121 or may perform merge actuation 13, they are sent to a merge unit 128. A selector (selector) is contained in the color space converter 125, and the raster print data from which the color space of loss nature or no losing nature was changed are sent to the output of the loss nature of the color space converter 125 connected to the print-data input of the loss nature of the merge unit 130, and no losing nature, or no losing nature.

[0075] Since the piece manager 124 of a page manages raster print data, the color space conversion

actuation 3, 9, and 12 of the print-data processing pipeline 15, the merge actuation 13, and the halftone actuation 14 receive the stream of raster print data which has a fixed format. For example, in the situation that the piece element of a page of one loss nature or no losing nature is generated by the image-processing actuation 1, as for the piece element of a page of corresponding loss nature and no losing nature, the piece manager 124 of a page generates the piece of the corresponding loss nature or no losing nature of blank print data of a page to a color space transducer 125 so that the pass top of loss nature and no losing nature may be sent (it was suitable like). Furthermore, in the case where the piece element of a page of one loss nature or no losing nature is generated by the image-processing actuation 1, the piece manager 124 of a page generates the corresponding merge plane formed in the same bit corresponding to each pixel of the piece element of a page. Furthermore, in the case where the same halftone processing is applied to each pixel of the piece of a page, the piece manager 124 of a page generates the halftone plane corresponding to the piece of a page. The piece manager 124 of a page sends merge data and halftone data to the merge unit 128 through separate cutting tool width-of-face pass.

[0076] The piece manager 124 of a page receives the raster print data of the loss nature elongated through the separate cutting tool width-of-face channel, and no losing nature. If compression loss nature raster print data are elongated by loss nature compression / stretcher 126, the block of 8 pixel x8 pixel as a result of each dimension of color space will be kept by three 64-byte buffers contained in loss nature compression / stretcher 126. The piece manager 124 of a page receives the cutting tool width-of-face stream of the loss nature raster print data sent from three output buffers. The cutting tool of the elongation loss nature raster print data corresponding to each pixel is seen off from the output buffer of loss nature compression / stretcher 126 as three cutting tools which has 1 byte corresponding to each dimension of color space and who continued. For example, the 1st of three cutting tools who continued corresponds to the "Y" element, the 2nd corresponds to the "Cb" element, and the 3rd corresponds to the "Cr" element. The piece manager 124 of a page assembles three corresponding cutting tools of loss nature raster print data about a pixel, and sends them to the color space converter 125 as a 24-bit width-of-face stream of loss nature raster print data.

[0077] The piece manager 124 of a page receives no elongation losing nature raster print data as a cutting tool width-of-face stream of three consecutive bytes about each nothing loss nature pixel. For example, the 1st of three consecutive bytes corresponds to the "R" element, the 2nd corresponds to the "G" element, and the 3rd corresponds to the "B" element. Merge data and halftone data are interleaved by no losing nature raster print data sent from no losing nature compression / stretcher 127 (interleave, insertion). The cutting tool of no losing nature raster print data corresponding to one line of the printed page is seen off, and an interleave is performed so that the required byte count of the merge data corresponding to Rhine of no losing nature raster print data and halftone data may follow this. The piece manager 124 of a page extracts merge data and halftone data, and sends these to the merge unit 128 through a separate channel. The piece manager 124 of a page assembles three corresponding cutting tools of no losing nature raster print data about a pixel, and sends these to the color space converter 125 as a 24-bit width-of-face stream of no losing nature raster print data.

[0078] In some cases, video DMA 122 may bypass a compressive step and raster print data may be sent to the piece manager 124 of a direct page. Without carrying out storage to compression and the

compression raster printing data memory 6, this is made, when the page of raster print data is sent to the print engine 16. In this case, raster print data flow through the print-data processing pipeline 15 linearly, without using what kind of various feed back passes. Moreover, in order to keep in memory the feed back pass top which raster print data can also be sent to the piece manager 124 of a page from the direct video DMA 122, and passes along processing actuation of the next print-data processing pipeline 15 like color space conversion and halftone processing which continues and takes place, and passes along video DMA 122 in order to send to a suitable compressor return and after that, compression raster print data are returned to video DMA 122.

[0079] No losing nature raster print data are specified by 1 bit per dimension of the color space of each pixel, 2 bits, 4 bits, or 8 bits. Before sending no losing nature raster print data to the color space transducer 125, the piece manager 124 of a page changes a display (1 bit, 2 bits, and 4 bits) into 8 bits of full per dimension of each pixel display. The parameter of raster print data can be changed between the piece elements of a page the same with having the class from which the print-data processing pipeline actuation applied to each of the piece element of a page differs. The piece manager 124 of a page performs actuation required to distribute the seamless stream of the raster print data of a fixed format to the downstream (lower stream of a river) of the print-data processing pipeline actuation (which the difference in the format of the raster print data between the piece elements of a page is removed as it is seamless here, and mean that there is almost no fluctuation in the rate of the flow of the raster print data which pass along the print-data processing pipeline 15).

[0080] Giving many kinds of print-data processing actuation which is likely to be performed of specific examples illustrates better the print-data processing pipeline's 15 remarkable versatility realized by ASIC120. The 1st case where a page is printed using the raster print data expressed by the CMYK engine color specification is considered. Here, any compression/elongation actuation is not performed, but halftone processing is applied to raster print data. The raster print data of loss nature and no losing nature are transmitted at once from a system memory through a part for one color plane, and the PCI bus interface 121. Video DMA 122 is sent to the color space converter 125 constituted so that print data may be bypassed for the raster print data of loss nature and no losing nature through the piece manager 124 of a page. The merge unit 128 performs merge actuation 13 about the raster print data of loss nature and no losing nature, and sends the merged print data to the halftone unit 13 for the halftone actuation 14. The raster print data by which halftone processing was carried out are sent to the print engine 16 through the print engine interface 131. The print-data processing pipeline 15 consisted of this 1st case so that only the merge actuation 13 and halftone actuation 14 might be performed.

[0081] The print-data processing pipeline 15 consists of the 2nd case so that elongation actuation 10 and 11 may be performed about the compression raster print data of loss nature and no losing nature and the color space conversion actuation 12, the merge actuation 13, and halftone actuation 14 may be performed. The compression raster print data of loss nature and no losing nature are transmitted through the PCI bus interface 121 from a system memory. Video DMA 122 sends the compression raster print data of loss nature and no losing nature to compression/stretcher of the loss nature 126 and no losing nature 127 for elongation. Since the raster print data of no elongation losing nature and loss nature change into one color plane about each of four pass of the print engine 16, it is sent to the color space converter 125 by them. In order that the raster print-data output changed from the color space transducer

125 of the color space of loss nature and no losing nature may perform merge actuation 13, it is sent to the merge unit 128. Halftone actuation 14 is performed about the print-data output merged from the merge unit 128. The result of halftone actuation is returned to video DMA 122, in order to send to a system memory through the PCI bus interface 121. It generates 4 times continuously and this the motion of a series of generates print data about each color plane of the print engine 16. In this 2nd case, the print-data processing pipeline 15 returns the print data processed by the system memory, in order to print later. Although video DMA 122 needed to deal with all print data twice (it is [1 time and] 1 time during storage during loading), two or more actuation was performed about the print data of ASIC120 without use of a PCI bus, or interference of a microprocessor 132.

[0082] The print-data processing pipeline 15 consists of the 3rd cases so that no losing nature compression actuation may be performed color space conversion actuation, halftone actuation, and after that. No losing nature raster print data are transmitted through the PCI bus interface 121 from a system memory. Video DMA 122 receives conversion to the plane of the color space of delivery and the print engine 16 for no losing nature raster print data to the color space transducer 125 through the piece manager 124 of a page. The piece manager 124 of a page generates the corresponding loss nature raster print data and the merge print data of a blank. Blank loss nature raster print data are bypassed about the color space transducer 125. The merge unit 130 uses the merge print data generated by the piece manager 124 of a page, and performs merge actuation about the raster print data of no losing nature and loss nature. The halftone unit 130 performs halftone actuation 14 about the merged raster print data. The raster print-data output from the halftone unit 130 is sent to video DMA 122. Video DMA 122 sends raster print data to no losing nature compression / stretcher 127 after that for no losing nature compression. Compression raster print data are returned to Video DMA, and are sent to a system memory through the PCI bus interface 121 for storage. It generates 4 times continuously and this the motion of a series of generates print data about each color plane of the print engine 16. In contrast with the 2nd case, the sequence of the performed print-data processing actuation was changed. It illustrates that this can be constituted so that the print-data processing pipeline 15 may optimize processing of print data.

[0083] A print-data processing pipeline's conventional technique was realized using the microprocessor for general, and various color space conversion, compression/elongation, merge, and halftone actuation have been performed. The speed of the microprocessor which performs firmware which performs these actuation is slower than what is attained using exclusive hardware. Furthermore, using the microprocessor for general, in order to finish these actuation produces the overhead of data transfer remarkable for the middle print data which transmit between a microprocessor and memory through a system bus. That the transfer through a system bus is very large reduces remarkably a print-data processing pipeline's throughput realized using the microprocessor for general.

[0084] It realized using the integrated circuit of dedication and the print-data processing pipeline has attained a print-data processing pipeline's specific function. Actuation like color space conversion, and compression/elongation has been realized by each integrated circuit. However, in addition, the microprocessor for general needs to move print data for between various integrated circuits and memory, when actuation of a print-data processing pipeline is performed. Although this offers some amelioration in the processing time by use of exclusive hardware, in addition, the overhead of data transfer is the

constraint factor of performance. Implementation of the print-data processing pipeline 15 of ASIC120 avoids the overhead of this data transfer by performing two or more pipeline actuation in ASIC120. [0085] The desirable operation gestalt of the merge unit 130 is chosen from the raster printing data stream of loss nature and no losing nature, forms one data stream using a merge bit, and reconfigures the first image. However, the technique used in the merge unit 130 should understand that it is effective also in combining many data streams from two using two or more merge bits about each unit of the data merged in the same way. Furthermore, the raster printing data stream of loss nature and no losing nature is an option, and the merged data stream can include the class of data different from the partition of raster print data or raster print data. For example, a raster printing data stream can be divided into two or more streams depending on application of the halftone actuation to the pixel of raster print data. These pixels of the raster print data which receive halftone actuation are compressed using the compression actuation optimized about the raster print data by which halftone processing was carried out. The pixel of the raster print data which did not receive halftone processing receives different compression actuation. Possible division of everything but a raster printing data stream is due to the number of bits used for expressing each color space of a pixel. For example, from the page field containing a color picture, a page field including the information on black and white lessens the bit per pixel, and is expressed. the field which has a page in the case of being the extremeest is full color per pixel display -- it can have 24 bits and other fields of a page can have 1 bit of binary black and white per pixel display. full color per pixel of raster print data -- processing separately a 24 bits stream and a 1 bit [per pixel of raster print data] stream makes it possible to optimize compression and halftone actuation about each of a stream. It is due to the resolution used for displaying a page that a raster printing data stream can be divided further. The field of the image which does not include so fine the description is printed in low resolution like 600dpi. The field of an image including the very fine description is printed in high resolution like 1200dpi. This case can be generated when the fixed gray scale field (gray scale region) which has a clear boundary exists. Processing separately the raster printing data stream of high resolution and the raster printing data stream of low resolution makes it possible to optimize halftone and compression actuation about each of a raster printing data stream.

[0086] Although the print-data processing pipeline's 15 desirable operation gestalt performs merge actuation 13 about digital data using the merge unit 130, it should be noticed also about it being possible to perform merge actuation to the front face of the photoconductor drum of the print engine 16 through the development (continuation pass) of the raster print data of the loss nature of the front face of a photoconductor drum (photoconductor drum) and no losing nature. This merge actuation is efficient OR actuation between the raster print data of loss nature and no losing nature, when a photoconductor drum is developed. Furthermore, instead of using a merge plane, merge actuation 13 can also be performed about digital data so that OR actuation may be performed between the raster print data of loss nature and no losing nature and the raster print data which should be chosen from these streams may be specified.

[0087] Drawing 4 expresses the actuation performed by the merge unit 128 by a diagram. Sections 20-23 are a part of SRAM line buffers 129, respectively, and each section corresponds to two buffers and holds one of four sorts of the printing data stream included in the merge unit 128. The thing whose buffering is the need is because [performing merge actuation 13] is the need about sufficient print data,

without being able to use it from each of a printing data stream by the way, and lowering the throughput of the raster print data of the merge unit 128. Since the merge unit 128 processes each merge bit, the cutting tool of raster print data has to be able to use it from both printing data streams of no losing nature or loss nature in order to assign the pixel currently processed. Furthermore, a halftone bit must be able to use it in order to maintain spatial relations with the pixel currently processed. When it is continuously put on the page by which a pixel is printed about no losing nature printing data stream, sufficient byte count for no losing nature line buffer 21 is kept so that the two whole scanning line of a print engine may be covered. By keeping two lines of no losing nature raster print data to two buffers which form no losing nature line buffer 21, no losing nature raster print data are movable through no losing nature line buffer 21 with the way of "ping-pong", and maintain high throughput.

[0088] Actuation of the data compression algorithm about the loss nature raster print data in the print-data processing pipeline's 15 desirable operation gestalt operates by dividing the pixel of the piece element of a loss nature page in order to form the cel used for a compression scan. Width of face is [8 pixels and the height of the size of the cel chosen with a print-data processing pipeline's desirable operation gestalt] 8 pixels. If elongation is performed by compression loss nature raster print data, although it corresponds to a block with a height of 8 pixels by width of face of 8 pixels, sufficient cutting tool of raster print data will be generated as an output of the loss nature elongation actuation 11. Since elongation of compression loss nature raster print data is made for every block in order to ensure sufficient buffering of the loss nature raster print-data output from the color space transducer 125, the loss nature line buffer 20 has the capacity which keeps the block of two lines. The block of two lines is equivalent to the sufficient number of pixels, and covers the whole scanning line of the print engine of 16. By keeping the block of two lines of loss nature raster print data to two buffers which form the loss nature line buffer 20, loss nature raster print data can be moved by the way of "ping-pong" through a loss nature line buffer, and maintain a high throughput. The merge line buffer 22 and the halftone line buffer 23 process the 2 scanning lines of a pixel including sufficient merge bit and a halftone bit, respectively. Therefore, the memory capacity of the merge line buffer 22 is 1/8 of no losing nature line buffer 21, and the memory capacity of the line buffer 23 of halftone is 1/4 of no losing nature line buffer 21.

[0089] The input of the printing data multiplexer 24 is connected to the output of the loss nature line buffer 20 and no losing nature line buffer 21. The output of the merge line buffer 22 controls selection of input raster print data for the output of the printing data multiplexer 24. Based on the situation of the merge bit of a pixel of having received processing, the printing data multiplexer 24 chooses raster print data about the pixel currently processed from either the loss nature raster printing data stream or no losing nature raster printing data stream. Since the raster printing data stream merged from the output of the printing data multiplexer 24 is the combination of two input raster printing data streams, the spatial synchronization between the pixels of the first image is maintained by the reconfigured image. The spatial synchronization for every pixel to the raster print data with which the halftone bit before assigned to the pixel to which both raster printing data streams of no losing nature and loss nature correspond existed when a halftone bit went into the merge unit 128 is maintained.

[0090] Drawing 5 is the high-level block diagram of the merge unit 128. The loss nature raster printing data stream 30 and no losing nature raster printing data stream 31 which consist of a 8-bit cutting tool's

stream, respectively are kept by the loss nature input buffer 32 and no losing nature input buffer 33, respectively. The stream of the merge bit 34 formed in the stream which consists of the cutting tool whose each is 8 bits, and the stream of the halftone bit 35 are kept by the merge input buffer 36 and the halftone input buffer 37, respectively. When a printing data stream is sent from input buffers 32, 33, 36, and 37, input buffers 32, 33, 36, and 37 are designed so that each of these printing data streams may be changed into 32-bit width of face from 8-bit width of face.

[0091] The color space transducer interface controller 38 manages a transfer of four printing data streams which pass along input buffers 32, 33, 36, and 37 including an input-buffer control function. This control function receives a handshaking signal from delivery and a converter 125 for a handshaking signal (handshaking signals) to the color space converter 125. These handshaking signals control the flow of the print data from the color space transducer 125 to input buffers 32, 33, 36, and 37. From these handshaking signals, the color space converter INTAESU controller 38 generates an enabling signal (enable signals), and it permits that input buffers 32, 33, 36, and 37 load the suitable class (loss nature, no losing nature, halftone, merge) of print data to one to which it corresponds of the input buffers 32, 33, 36, and 37. The handshaking signal received from the color space converter 125 includes the signal which shows the time when the raster print data of loss nature and no losing nature are usable although transmitted to the input buffer of the loss nature 32 and no losing nature 33. Moreover, the color space transducer interface controller 38 receives the handshaking signal with which a merge and halftone print data show the time at which it can transmit to the input buffer of merge 36 and halftone 37. Moreover, the handshaking signal sent to the color space transducer 125 and the piece manager 124 of a page from the color space transducer interface controller 38 includes the signal which shows the time of being ready for each of input buffers 32, 33, 36, and 37 receiving input print data.

[0092] The 32-bit width-of-face data streams 40, 41, and 42 from each of input buffers 32, 33, 36, and 37 and 43 outputs are sent to the input-buffer multiplexer 39. The color space transducer interface controller 38 controls the flow of the printing data stream from the input buffers 32, 33, 36, and 37 which pass along the input-buffer multiplexer 39. In order to transmit the control signal 47 generated by the color space converter interface controller 38 to the SRAM interface controller 44, it chooses one of the 32-bit width-of-face data streams 40, 41, 42, and 43. In order that the situation device (state machine) (not shown separately) included in the color space converter interface controller 38 may generate a control signal 47 and which of the printing data stream from input buffers 32, 33, 36, and 37 may transmit it to the SRAM interface controller 44, it judges whether it is chosen by the input-buffer multiplexer 39.

[0093] The SRAM interface controller 44 and the color space converter interface controller 38 generate the required handshaking signal 45, respectively, and transmit print data to the SRAM interface controller 44 through the input-buffer multiplexer 39. The SRAM interface controller 44 generates a handshaking signal, and it is shown in the color space converter interface controller 38 that there is preparation with which the SRAM interface controller 44 receives print data. The color space transducer interface controller 38 offers the control signal which identifies the class (loss nature, no losing nature, halftone, merge) of print data sent from the input-buffer multiplexer 39. The print data received by the SRAM interface controller 44 are kept by the SRAM line buffer 129. The SRAM line buffer 129 is divided into the section corresponding to each of the class (loss nature, no losing nature, halftone,

merge) of print data. The line buffers 20-23 of the SRAM line buffer 129 correspond to these which are shown in drawing 4 . Storage of the print data of the SRAM line buffer 129 guarantees that it can be used for performing merge actuation which continued sufficient print data. The print data received by the SRAM interface controller 44 are kept into the suitable section of the SRAM line buffer 129, as judged by the control signal 45 from the color space converter interface controller 38. Since the print data kept by the SRAM line buffer 129 are loaded to the SRAM interface controller 44, they can use print data for performing merge actuation 13. Print data are kept by the SRAM line buffer through the bus 49 of one 32 bit width of face of bidirectional, and are loaded from there. Although this bus 49 is shown in drawing 5 as two unidirectional buses, one bi-directional bus is used for the desirable operation gestalt of the merge unit 128. The SRAM interface controller 44 generates the address and accesses the memory location of the SRAM line buffer 129 through an address bus 50.

[0094] It equips performing merge actuation with the print data loaded to the SRAM interface controller 44 from the SRAM line buffer 129, and they are transmitted to output buffers 51, 52, 53, and 54. Output buffers 51, 52, 53, and 54 contain the loss nature output buffer 51, no losing nature output buffer 52, the merge output buffer 53, and the halftone output buffer 54. The halftone output buffer 54 and the merge output buffer 53 are formed from a shift register, respectively. using a shift register -- the 32-bit WORD of halftone print data and merge print data -- a pair of bit -- or (halftone print data) it becomes possible to divide (merge print data) at each bit, and these are compared with the raster print data corresponding to the usually related pixel. A transfer of the print data from a SRAM interface controller to output buffers 51, 52, 53, and 54 is managed by the control signal 56 between the halftone interface controller 55 and the SRAM interface controller 44. A control signal 56 includes the signal which shows the time of having the print data which can be used for the SRAM interface controller 44 transmitting to output buffers 51, 52, 53, and 54 including the signal which shows the time of there being preparation whose each of output buffers 51, 52, 53, and 54 receives the class (loss nature, no losing nature, halftone, merge) to which print data correspond.

[0095] The halftone interface controller 55 generates a control signal 57, and controls a transfer of the print data which pass along output buffers 51, 52, 53, and 54. The input printing data stream through which output buffers 51, 52, 53, and 54 pass, respectively is 32-bit width of face. The raster printing data stream output of the loss nature from the loss nature raster print-data output buffer 51 and no losing nature raster print-data output buffer 52 and no losing nature is 8-bit width of face, respectively. A control signal 57 permits that the class of suitable print data is loaded to the corresponding output buffers 51, 52, 53, and 54 including an enabling signal. A multiplexer is contained in each of the raster print-data output buffer of the loss nature 51 and no losing nature 52 (not shown in drawing 5), and it chooses one of four 8-bit cutting tools to the printing data multiplexer 24 contained in the raster printing data stream of the loss nature of 32-bit width of face, and no losing nature for a transfer. A control signal 57 includes a signal required to control the multiplexer which are a part of loss nature output buffer 51 and no losing nature output buffer 52. The control signal 57 generated by the halftone interface controller 55 includes a signal required to load the halftone and the merge data stream of 32-bit width of face to the corresponding halftone data output buffer 54 and the corresponding merge data output buffer 53. As mentioned above, each of the halftone data output buffer 54 and the merge data output buffer 53 contains a shift register. Furthermore, as each of the halftone data output buffer 54

and the merge data output buffer 53 was judged by the signal included in a control signal 57 including a multiplexer (not shown in drawing 5), either a right shift output or a left shift output is chosen.

Moreover, this signal controls the direction from which a shift register moves merge data and halftone data. It is required to control the direction of the shift register to which merge data and halftone data are moved because of double-sided mode (duplex mode) printing. In double-sided mode printing, the sequence that raster print data are distributed to the print engine 16 becomes hard flow between printings on the side front and background of a page on the basis of the top and the bottom of a page.

[0096] The printing data multiplexer 24 receives the raster printing data stream of the loss nature of cutting tool width of face, and no losing nature as an input. The 1-bit width-of-face merge data stream output from the merge output buffer 53 controls one selection in an input raster printing data stream for the output from the merge unit 128. When moved through the various buffers of the merge unit 128, the merge actuation 13 performed by the printing data multiplexer 24 reconfigures the first page (it is original) correctly for every pixel by maintaining the relative sequence of the raster print data of the relative sequence of the merge bit of a merge data stream, the loss nature of each raster printing data stream, and no losing nature. To this contractor of the technical field of a digital design, probably, the knowledge which designs a logical circuit required to perform the function of the printing data multiplexer 24 is owned, after understanding this specification.

[0097] The SRAM line buffer 129 enables establishment synchronous [between the raster print data of loss nature and no losing nature, merge data, and halftone data / time and spatial]. Since elongation loss nature raster print data are elongated with a block with a height of 8 pixels by 8-pixel width of face, elongation loss nature raster print data are distributed to the merge unit 128, from the synchronization to other print data, shift and are kept by the SRAM line buffer 129. The synchronization with the time and spatial approach (it separates) which print data move from the SRAM line buffer 129 is established. It should be cautious of the implementation of a merge unit which does not need use of the SRAM line buffer 129 being possible. If the raster printing data stream and merge data of loss nature and no losing nature are given to a merge unit by the approach which synchronized in time and spatially, the need for buffering of the printing data stream in the interior of the merge unit 128 will be removed. In order for implementation of a required merge unit to receive the printing data stream which synchronized in time and spatially, if required, buffering of a printing data stream will be performed in the exterior of a merge unit. Furthermore, time and the merge unit which functions by spatial synchronization using a printing data multiplexer between the raster printing data stream of loss nature and no losing nature and a merge data stream are simplified remarkably.

[0098] The merge data output buffer 53 changes the cutting tool width-of-face stream of merge data into the 1-bit width-of-face stream (a corresponding pixel is expressed) which synchronized with the raster print data of loss nature or no losing nature. Although 1 bit is used in order to choose the desirable operation gestalt of the merge unit 128 from the cutting tool of the raster print data of loss nature and no losing nature, two or more bit width-of-face stream can also be used for choosing from more raster printing data streams than two, and from 8 bits, a raster printing data stream can use many and can also express each dimension of color space.

[0099] Drawing 6 is the conceptual diagram by which input-side arrangement of the loss nature input buffer 32 was simplified, and expresses each actuation of input buffers 32, 33, 36, and 37. The typical

loss nature input buffer 32 is formed from eight 8-bit data latches 60-67. The 8-bit input disadvantage degeneration raster printing data stream 68 is sent to eight "DATA" inputs of eight data latches 60-67. Eight 3 input AND gates 69-76 connected to eight data latches' 60-67 "EN" input decode a control signal (generated by the color space transducer interface controller 38), and choose one of the eight data latches 60-67 for loading of the cutting tool of an input printing data stream. Decode logic is some color space converter interface controllers 38. As shown in drawing 6 , eight data latches 60-67 are divided into two banks where each has four data latches. Decode of a control signal is made so that four cutting tools of a printing data stream transmitted continuously may be loaded to one of the two of two banks and four cutting tools next to a printing data stream transmitted continuously may be loaded to another side of two banks. Loading to input buffers 32, 33, 36, and 37 maintains the flow of the print data which pass along input buffers 32, 33, 36, and 37 so that 4 bytes of group of a printing data stream transmitted continuously may be loaded by turns to one of two banks of a data latch. While one of the two of two banks is loaded with input print data, a transfer of the print data from another side of two banks occurs. [0100] Drawing 7 is the conceptual diagram by which output side arrangement of the loss nature input buffer 32 was simplified, and expresses each actuation of input buffers 32, 33, 36, and 37. Eight data latches' 60-67 "OE" input is connected to fixed logic level, and the loss nature raster print data kept by the loss nature input buffer 32 are outputted by the data latches 60-67 by the rising edge of each clock cycle. Each four 8-bit output of two banks of the loss nature input buffer 32 forms the print data of two 32-bit width of face. These two 32-bit width-of-face printing data streams are connected to the input of a multiplexer 70. 1 of these two and a 32-bit width-of-face printing data stream are chosen for the input to the input-buffer multiplexer 39. Selection uses the control signal generated by the color space converter interface controller 38, and is performed by the multiplexer 70. It detects whether this control signal is ready to have the print data which which is full (full) and are contained in being sent to the input-buffer multiplexer 39 among input buffers 32, 33, and 36 and two banks of 37. Since one of 4 bytes of groups kept by each of 2 banks is chosen by turns, a transfer of the print data from input buffers 32, 33, 36, and 37 maintains the flow of the print data which pass along input buffers 32, 33, 36, and 37, and maintains the flow of the print data to the input-buffer multiplexer 39. While 4 bytes of print data are transmitted by one of the two of two banks, 4 bytes of loading of the print data to another side of two banks occurs.

[0101] Drawing 8 shows the conceptual diagram by which input-side arrangement of the loss nature output buffer 51 and no losing nature output buffer 52 was simplified. Each of the loss nature 51 and no losing nature 52 output buffer includes two banks of four 8 bit-data latches 80-95. The loss nature raster printing data stream of the 32-bit width of face inside the merge unit 128 and no losing nature raster printing data stream are divided into the raster printing data stream of four 8-bit width-of-face loss nature or no losing nature, respectively, and are sent to each data latches' 80-95 each "DATA" input. Each "EN" input of the data latches 80-95 controls loading to the data latches 80-95 of the raster print data of loss nature and no losing nature. The four AND gates 96-99 are some halftone interface controllers 55, decode the control signal generated by the halftone interface controller 55, and control the data latches' 80-95 loading. While the raster print data of loss nature and no losing nature are loaded to one pair of the data latches' 80-95 banks, the raster print data of loss nature and no losing nature are sent from one pair of other banks of the data latches 80-95. Use of this buffering approach maintains the

flow of the print data which pass along loss nature and no losing nature output buffers 51 and 52.

[0102] Drawing 9 is the conceptual diagram which output side arrangement of one pair of banks of the data latches 80-83 about the loss nature 51 and no losing nature 52 output buffer, the loss nature of 88-91, and no losing nature simplified. The data latches 84-87 of the loss nature 51 and no losing nature 52 output buffer and the output side of other two banks of 92-95 are realized similarly. The loss nature 51 and no losing nature 52 output buffer change the raster printing data stream of 32-bit width-of-face input disadvantage degeneration and no losing nature into the raster printing data stream of the loss nature of 8-bit width of face, and no losing nature. Eight data latches 80 - the "OE" input of 83, 88-91 are connected to fixed logic level, and the raster print data of these data latches 80-83, the loss nature kept by 88-91, and no losing nature are outputted by the rising edge of each clock cycle. Multiplexers 100 and 101 are used for choosing one of 83 and 88 to 91 to 8-bit outputs from each data latch 80 to the printing data multiplexer 24 for an input. Since each data latch 80-83 and a 88-91 to 8-bit output are chosen by each multiplexer 100 and 101, the relative sequence of the 8-bit cutting tool of the raster printing data stream of each loss nature and no losing nature is maintained. This needs the merge actuation 13 performed by the printing data multiplexer 24 to reconfigure the first page correctly. The signal used for controlling multiplexers 100 and 101 is generated by the halftone interface controller 55. The stream of the merge bit output by the merge output buffer 53 controls selection of the 8-bit cutting tool from the raster printing data stream of loss nature and no losing nature by the print-data multiplexer 24. The raster print data taken out from the print-data multiplexer 24 are 8-bit width of face, and consist of the raster printing data stream of the merged loss nature and no losing nature.

[0103] Drawing 10 is the conceptual diagram by which the halftone output shift register 110 and the merge output shift register 111 were simplified. The 32-bit width-of-face halftone printing data stream 112 and the 32-bit width-of-face merge printing data stream 113 are loaded to each output shift register 110 and 111. The signal to the each "LD" input of output shift registers 110 and 111 from the halftone interface controller 55 controls loading of halftone and merge print data. By use of shift registers 110 and 111, since it becomes possible to divide into the stream of the print data of 2-bit width of face and 1-bit width of face the cutting tool by whom halftone print data and merge print data were adjusted, respectively, the relative sequence of these two data streams is maintained about the raster printing data stream of loss nature and no losing nature.

[0104] The AND gates 114 and 115 are included in the halftone interface controller 55, and control the shift-out of the merge print data from each shift register 110 and 111, and halftone print data. If the "EN" input of shift registers 110 and 111 is asserted, each register will carry out the shift-out of merge print data and the halftone print data. Depending on the level of the "DIR" input to shift registers 110 and 111, the shift-out of a merge and the halftone print data is carried out from "SLD" or the "SRD" output by the rising edge of a clock. The "DIR" input controls a merge of registers 110 and 111, and the direction of the shift of halftone print data so that the relative sequence of a merge and halftone print data is maintained on the basis of the raster printing data stream of loss nature and no losing nature. When formed in each 32-bit width-of-face printing data stream inside the merge unit 128, depending on how a merge and a halftone bit are put in order, the "DIR" input is controlled to the 32-bit most significant and a least significant bit to maintain the spatial synchronization to the printing data stream of loss nature and no losing nature.

[0105] The halftone printing data stream by which the shift-out was carried out from the shift register 110 is 2-bit width of face, and the merge data stream by which the shift-out was carried out from the shift register 111 is 1-bit width of face. Multiplexers 116 and 117 are controlled by the same signal connected to the "DIR" input of shift registers 110 and 111, and since it outputs from multiplexers 116 and 117, they choose "SRD" or the "SLD" output of shift registers 110 and 111. It connects with the control input of the print-data multiplexer 24, and the 1-bit width-of-face stream of the merge print-data output from a multiplexer 117 is chosen from the raster printing data stream of loss nature or no losing nature for every pixel. The output print data from the merge unit 128 include the halftone printing data stream output of the 2-bit width of face from a multiplexer 116 including the raster printing data stream output into which the 8-bit width of face from the buffer multiplexer 24 was merged.

[0106] although some operation gestalten of this invention were illustrated and explained -- this contractor of the technical field concerned -- this invention ** -- probably, it will be clear that various amelioration [be / no detached building *****] can be made.

[0107] This invention includes the following embodiment as an example.

(1) He is a data-processing pipeline (200 15) for processing separately two or more data streams (201 202) formed from two or more data elements, respectively. The data-processing pipeline who uses the merge data stream (209) which is defined by said two or more data streams (201 202), and is formed from the merge data element corresponding to said data element. It has the 1st input which receives the 1st data stream (201) of two or more of said data streams (201 202). The 1st pipeline processing unit which generates the data stream (205) from which 1st conversion was performed about said data element of this 1st data stream, and the 1st was changed (203). It has the 2nd input which receives the 2nd data stream (202) of two or more of said data streams (201 202). The 2nd pipeline processing unit which generates the data stream (206) from which 2nd conversion was performed about said data element of this 2nd data stream, and the 2nd was changed (204). It is constituted so that said data stream (205) from which the 1st was changed, said data stream (206) from which the 2nd was changed, and said merge data stream (209) may be received. The data-processing pipeline having the merge unit (207 128) by which said data stream from which the 1st was changed, and said data stream from which the 2nd was changed are merged to an output data stream (208) using said merge data stream.

[0108] (2) In the data-processing pipeline (200 15) in the above (1) Said data element of said 1st data stream (201) of two or more of said data streams (200 201) No losing nature data and no compression losing nature data are included alternatively. Said 2nd data stream (202) of two or more of said data streams Said merge data element contains merge data including loss nature data and compression loss nature data alternatively. Said 1st pipeline processing unit (203) contains no losing nature compression / stretcher (127) which has no losing nature elongation output and no losing nature compression output. This no losing nature compression / stretcher include the function which generates reception elongation merge data for the function which generates reception and compression merge data for said merge data, and said compression merge data. Said data stream (205) from which the 1st was changed No compression losing nature data and no elongation losing nature data are included alternatively. Said 2nd pipeline processing unit The loss nature compression / stretcher (126) which has a loss nature elongation output and a loss nature compression output are included. Said data stream from which the 2nd was changed contains compression loss nature data and elongation loss nature data alternatively.

Said output data stream (208) contains no losing nature and loss nature data which were merged. Said merge unit (207 128) Said no losing nature data, said loss nature data, and said merge data Reception, The data-processing pipeline of the above (1) including the function which generates said no losing nature and loss nature data which were merged, and generates reception, said no merged losing nature, and loss nature data for said no elongation losing nature data, said elongation loss nature data, and said elongation merge data.

[0109] (3) In the data-processing pipeline (200 15) in the above (2) It connects with said no losing nature compression output and said loss nature compression output. Said no losing nature, Said no compression losing nature, said no elongation losing nature data, said loss nature, said compression loss nature, It has said elongation loss nature data and said merge, said compression merge, and the 3rd input that receives at least one of said elongation merge data. Said no losing nature, said no compression losing nature, said no elongation losing nature data, and said merge, At least one of said compression merge and said elongation merge data to said no losing nature compression / stretcher (127) Delivery, said loss nature, said compression loss nature, and said elongation loss nature data -- with the Direct-Memory-Access controller (122) constituted so that at least one might be sent to said loss nature compression / stretcher (126) It connects with said no losing nature elongation output, said loss nature elongation output, and said Direct-Memory-Access controller. With the piece manager of a page who has the merge data output connected to no 1st losing nature output, the 1st loss nature output, and said merge unit (128) (124) It is the color space converter (125) which has no 2nd losing nature output and the 2nd loss nature output which were connected to said no 1st losing nature output and said 1st loss nature output, and were connected to the merge unit. The data-processing pipeline of the above (2) equipped with the color space transducer in which each of said the 1st and 2nd conversion includes color space conversion.

[0110] In the data-processing pipeline in the above (3) (4) Said color space converter (125) Said no losing nature data which pass along said color space transducer, said no elongation losing nature data, The function which bypasses said loss nature data and said elongation loss nature data is included. Said color space converter (125) In order to choose from said no losing nature data and said loss nature data, and in order to choose from said no elongation losing nature data and said elongation loss nature data, The selector which has the selector output connected to said Direct-Memory-Access controller (122) is included. Said no losing nature and loss nature data No losing nature raster print data and loss nature raster print data are included, respectively for the use in a printing system. The data of said no merged losing nature and loss nature No losing nature and loss nature raster print data which were merged are included. Said piece manager (124) of a page For the next processing actuation, said no losing nature raster print data, said no elongation losing nature raster print data, Said printing system is the data-processing pipeline of the above (3) containing an electrophotography method (electrophotographic) printing system (16) including the function which formats at least one of said loss nature raster print data and said the elongation loss nature raster print data.

[0111] (5) He is a print-data processing pipeline (200 15) for processing separately two or more printing data streams (201 202) formed from two or more print-data elements, respectively. It is set by said two or more printing data streams (201 202). The print-data processing pipeline who uses the merge data stream (209) formed from the merge data element corresponding to said print-data element It has the 1st

input which receives the 1st printing data stream (201) of two or more of said printing data streams (201 202). this -- 1st conversion being performed about said print-data element of the 1st printing data stream, and with the 1st pipeline processing unit (203) which generates the printing data stream (205) from which the 1st was changed It has the 2nd input which receives the 2nd printing data stream (202) of two or more of said printing data streams (201 202). this -- 2nd conversion being performed about said print-data element of the 2nd printing data stream, and with the 2nd pipeline processing unit (204) which generates the printing data stream (206) from which the 2nd was changed It is constituted so that said printing data stream (205) from which the 1st was changed, said printing data stream (206) from which the 2nd was changed, and said merge data stream (209) may be received. The print-data processing pipeline having the merge unit (207 128) by which an output printing data stream (208) is merged for said printing data stream from which the 1st was changed, and said printing data stream from which the 2nd was changed using said merge data stream.

[0112] (6) In the print-data processing pipeline (200 15) in the above (5) Said merge data element contains merge data and elongation merge data alternatively. Said data element of said 1st stream of two or more of said printing data streams No losing nature raster print data and no compression losing nature raster print data are included alternatively. Said data element of said 2nd stream of two or more of said printing data streams Loss nature raster print data and compression loss nature raster print data are included alternatively. Said output printing data stream (208) The raster print data of no merged losing nature and loss nature are included. Said 1st pipeline processing unit (203) Said no compression losing nature raster print data are generated from said no losing nature raster print data. No losing nature compression / stretcher (127) which generates no elongation losing nature raster print data from said no compression losing nature raster print data are included. Said 2nd pipeline processing unit (204) Said compression loss nature raster print data are generated from said loss nature raster print data. The data stream (205) into which said 1st [the] was changed from said compression loss nature raster print data including the loss nature compression / stretcher (126) which generates elongation loss nature raster print data Said data stream (206) from which the 2nd was changed including said no losing nature raster print data, said no compression losing nature raster print data, and said no elongation losing nature raster print data alternatively Said loss nature raster print data, said compression loss nature raster print data, and said elongation loss nature raster print data are included alternatively. Said no losing nature compression / stretcher (127) No losing nature elongation output and no losing nature compression output are included. Said merge data Reception, Said loss nature compression / stretcher (126) are the print-data processing pipeline of the above (5) included in a loss nature elongation output and a loss nature compression output including the function which generates the reception aforementioned elongation merge data for the function which generates compression merge data, and said compression merge data.

[0113] (7) The print-data processing pipeline in the above (6) is further connected to said no losing nature compression output and said loss nature compression output. Said no losing nature, said no compression losing nature, said no elongation losing nature raster print data, Said loss nature, said compression loss nature, said elongation loss nature raster print data, and said merge, It has said compression merge and the input which receives at least one of said elongation merge data. Said no losing nature, said no compression losing nature, said no elongation losing nature data, and said merge,

It is constituted so that at least one of said compression merge and said elongation merge data may be sent to said no losing nature compression / stretcher (127). Said loss nature, said compression loss nature, and the Direct-Memory-Access controller constituted so that at least one of said elongation loss nature raster print data might be sent to said loss nature compression / stretcher (126) (122); The processor which generates said no losing nature raster print data, said loss nature raster print data, and said merge data from print data, and is sent to said input of a Direct-Memory-Access controller (132), It is outputted to said no losing nature elongation output, said loss nature elongation output, and said Direct-Memory-Access controller. It has no 1st losing nature output, the 1st loss nature output, and the merge data output connected to said merge unit. Said no losing nature raster print data, said loss nature raster print data, and said merge data, With the piece manager of a page including the function formatted for the next processing actuation of at least one of said no elongation losing nature raster print data, said elongation loss nature raster print data, and said elongation merge data It has no 2nd losing nature output and the 2nd loss nature output which were connected to said no 1st losing nature output and said 1st loss nature output, and were connected to said merge unit. Said no losing nature raster print data which pass along said color space transducer, said no elongation losing nature raster print data, In order to choose from said no losing nature raster print data and said loss nature raster print data including the function which bypasses said loss nature raster print data and said elongation loss nature raster print data, Or in order to choose from said no elongation losing nature raster print data and said elongation loss nature raster print data, The print-data processing pipeline of the above (6) equipped with the color space transducer containing the selector which has the selector output connected to said Direct-Memory-Access controller.

[0114] (8) Generate two or more printing data streams (201 202) which are printers and are formed from two or more print-data elements, respectively. The processor which generates the merge data stream which is defined by said two or more printing data streams, and is formed from the merge data element corresponding to said print-data element (132), The Direct-Memory-Access controller constituted so that said two or more printing data streams might be received from said processor (122), It has the 1st input which receives the 1st printing data stream from said Direct-Memory-Access controller and said 1st pipeline processing unit (203 127) to the inside of two or more of said printing data streams. this -- 1st conversion being performed about said print-data element of the 1st printing data stream, and with the 1st pipeline processing unit which generates the printing data stream from which the 1st was changed It has the 2nd input which receives the 1st printing data stream from said Direct-Memory-Access controller and said 2nd pipeline processing unit (204 126) to the inside of two or more of said printing data streams. this -- 2nd conversion being performed about said print-data element of the 2nd printing data stream, and with the 2nd pipeline processing unit which generates the printing data stream from which the 2nd was changed It is constituted so that said printing stream from which the 1st was changed, said printing stream from which the 2nd was changed, and said merge data stream may be received. A printer equipped with the merge unit (207 128) by which the printing data stream from which said printing data stream from which the 1st was changed, and said 2nd [the] were changed is merged to an output printing data stream using said merge data stream.

[0115] In the printer in the above (8) (9) Said merge data element Merge data and elongation merge data are included alternatively. Said data element of said 1st printing stream of two or more of said printing

data streams. No losing nature raster print data and no compression losing nature raster print data are included alternatively. Said data element of said 2nd printing stream of two or more of said printing data streams. Loss nature raster print data and compression loss nature raster print data are included alternatively. Said output printing data stream. No losing nature and loss nature raster print data which were merged are included. Said 1st pipeline processing unit. Said no compression losing nature raster print data are generated from said no losing nature raster print data. No losing nature compression / stretcher which generates no elongation losing nature raster print data from said no compression losing nature raster print data are included. Said 2nd pipeline processing unit. Said compression loss nature raster print data are generated from said loss nature raster print data. Said data stream from which the 1st was changed including the loss nature compression / stretcher which generates elongation loss nature raster print data from said compression loss nature raster print data. Said data stream from which the 2nd was changed including said no losing nature raster print data, said no compression losing nature raster print data, and said no elongation losing nature raster print data alternatively. Said loss nature raster print data, said compression loss nature raster print data, and said elongation loss nature raster print data are included alternatively. Said no losing nature compression / stretcher (127) Including no losing nature elongation output and no losing nature compression output, receive said merge data and compression merge data are generated. Said loss nature compression / stretcher (126) are the printer of the above (8) including a loss nature elongation output and a loss nature compression output including the function which receives said compression merge data and generates said elongation merge data.

[0116] In the printer in the above (9) (10) Said no losing nature elongation output, It connects with said loss nature elongation output and said Direct-Memory-Access controller. It has the merge data output connected to no 1st losing nature output, the 1st loss nature output, and said merge unit. Said no losing nature raster print data, said no elongation losing nature raster print data, With the piece manager of a page including the function formatted for the next processing actuation of at least one of said loss nature raster print data, said elongation loss nature raster print data, said merge data, and said the elongation merge data. It has no 2nd losing nature output and the 2nd loss nature output which were connected to said no 1st losing nature output and said 1st loss nature output, and were connected to said merge unit. Said no losing nature raster print data which pass along said color space transducer, said no elongation losing nature raster print data, In order to choose from said no losing nature raster print data and said loss nature raster print data including the function which bypasses said loss nature raster print data and said elongation loss nature raster print data, And the printer of the above (9) equipped with the color space converter containing the selector which has the selector output connected to said Direct-Memory-Access controller in order to choose from said no elongation losing nature raster print data and said elongation loss nature raster print data.

[0117]

[Effect of the Invention] According to the data-processing pipeline by this invention, based on a data element, a compression ratio and the optimal combination of printing quality can be applied now. Moreover, printing quality can be further improved actuation in a print-data processing pipeline by *Lycium chinense* in a suitable location, and the required amount of memory can be decreased now.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] A print-data processing pipeline's high-level Fig.

[Drawing 2] The conceptual diagram of the flow of the print data which pass along a print-data processing pipeline's desirable operation gestalt in a color printer.

[Drawing 3] The hardware block diagram of a print-data processing pipeline's desirable operation gestalt.

[Drawing 4] The conceptual diagram of merge actuation.

[Drawing 5] The high-level block diagram of a merge unit.

[Drawing 6] The conceptual diagram of the input side of a loss nature input buffer.

[Drawing 7] The conceptual diagram of the output side of a loss nature input buffer.

[Drawing 8] The conceptual diagram of the input side of loss nature and no losing nature output buffer.

[Drawing 9] The conceptual diagram of the output side of loss nature and no losing nature output buffer.

[Drawing 10] The conceptual diagram of the shift register used for halftone and a merge printing data stream.

[Description of Notations]

15,200 Print-data processing pipeline

201 1st Data Stream

202 2nd Data Stream

205 Data Stream from which the 1st was Changed

206 Data Stream from which the 2nd was Changed

128 207 Merge unit

208 Output Data Stream

209 Merge Data Stream

[Translation done.]